

ASG-Manager Products™

Quick Reference

Version 2.5

Publication Number: MPR0500-25-QKREF

Publication Date: December 2000

The information contained herein is the confidential and proprietary information of Allen Systems Group, Inc. Unauthorized use of this information and disclosure to third parties is expressly prohibited. This technical publication may not be reproduced in whole or in part, by any means, without the express written consent of Allen Systems Group, Inc.

© 1998-2002 Allen Systems Group, Inc. All rights reserved.

All names and products contained herein are the trademarks or registered trademarks of their respective holders.

ASG Documentation/Product Enhancement Fax Form

Please FAX comments regarding ASG products and/or documentation to (239) 263-3692.

Company Name	Telephone Number	Site ID	Contact name

Product Name/Publication	Version #	Publication Date
Product:		
Publication:		
Tape VOLSER:		

Enhancement Request:

ASG Support Numbers

ASG provides support throughout the world to resolve questions or problems regarding installation, operation, or use of our products. We provide all levels of support during normal business hours and emergency support during non-business hours. To expedite response time, please follow these procedures.

Please have this information ready:

- Product name, version number, and release number
- List of any fixes currently applied
- Any alphanumeric error codes or messages written precisely or displayed
- A description of the specific steps that immediately preceded the problem
- The severity code (ASG Support uses an escalated severity system to prioritize service to our clients. The severity codes and their meanings are listed below.)
- Verify whether you received an ASG Service Pack for this product. It may include information to help you resolve questions regarding installation of this ASG product. The Service Pack instructions are in a text file on the distribution media included with the Service Pack.

If You Receive a Voice Mail Message:

- 1 Follow the instructions to report a production-down or critical problem.
- 2 Leave a detailed message including your name and phone number. A Support representative will be paged and will return your call as soon as possible.
- 3 Please have the information described above ready for when you are contacted by the Support representative.

Severity Codes and Expected Support Response Times

Severity	Meaning	Expected Support Response Time
1	Production down, critical situation	Within 30 minutes
2	Major component of product disabled	Within 2 hours
3	Problem with the product, but customer has work-around solution	Within 4 hours
4	"How-to" questions and enhancement requests	Within 4 hours

ASG provides software products that run in a number of third-party vendor environments. Support for all non-ASG products is the responsibility of the respective vendor. In the event a vendor discontinues support for a hardware and/or software product, ASG cannot be held responsible for problems arising from the use of that unsupported version.

Business Hours Support

Your Location	Phone	Fax	E-mail
United States and Canada	800.354.3578	239.263.2883	support@asg.com
Australia	61.2.9460.0411	61.2.9460.0280	support.au@asg.com
England	44.1727.736305	44.1727.812018	support.uk@asg.com
France	33.141.028590	33.141.028589	support.fr@asg.com
Germany	49.89.45716.222	49.89.45716.400	support.de@asg.com
Singapore	65.6332.2922	65.6337.7228	support.sg@asg.com
All other countries:	1.239.435.2200		support@asg.com

Non-Business Hours - Emergency Support

Your Location	Phone	Your Location	Phone
United States and Canada	800.354.3578		
Asia	65.6332.2922	Japan/Telecom	0041.800.9932.5536
Australia	0011.800.9932.5536	Netherlands	00.800.3354.3578
Denmark	00.800.9932.5536	New Zealand	00.800.9932.5536
France	00.800.3354.3578	Singapore	001.800.3354.3578
Germany	00.800.3354.3578	South Korea	001.800.9932.5536
Hong Kong	001.800.9932.5536	Sweden/Telia	009.800.9932.5536
Ireland	00.800.9932.5536	Switzerland	00.800.9932.5536
Israel/Bezeq	014.800.9932.5536	Thailand	001.800.9932.5536
Japan/IDC	0061.800.9932.5536	United Kingdom	00.800.9932.5536
		All other countries	1.239.435.2200

ASG Web Site

Visit <http://www.asg.com>, ASG's World Wide Web site.

Submit all product and documentation suggestions to ASG's product management team at <http://www.asg.com/asp/emailproductsuggestions.asp>.

If you do not have access to the web, FAX your suggestions to product management at (239) 263-3692. Please include your name, company, work phone, e-mail ID, and the name of the ASG product you are using. For documentation suggestions include the publication number located on the publication's front cover.

Contents

Preface	iii
About this Publication	iii
Publication Conventions	iv
1 Commands	1
2 Member Types	209
3 Procedures Language	295
EXECUTIVE-ROUTINE and SEXEC MEMBER-TYPES Syntax	295
User Assigned Variables	296
System Variables	296
Return Codes	298
EXPRESSIONS	299
Character Expressions	299
Numeric Expressions	299
Appendix A	
Name, Status, Time, and User-related Selection	301
Secondary Selection Clauses: Name-related-selection	301
Secondary Selection Clauses: Status-related-selection	302
Secondary Selection Clauses: Time and User-related-selection	303
Appendix B	
Common Clauses	305
Appendix C	
Features Table	309

Preface

This *ASG-Manager Products Quick Reference* provides syntax for all commands and the most commonly-used member types, not restricted to Administrators or Controllers.

Allen Systems Group, Inc. (ASG) provides professional support to resolve any questions or concerns regarding the installation or use of any ASG product. Telephone technical support is available around the world, 24 hours a day, 7 days a week.

ASG welcomes your comments, as a preferred or prospective customer, on this publication or on any ASG product.

About this Publication

This publication consists of these chapters:

- [Chapter 1, "Commands,"](#) provides an alphabetical list of the syntax for all non-restricted commands.
- [Chapter 2, "Member Types,"](#) provides an alphabetical list of the syntax for selected non-DBMS member types.
- [Chapter 3, "Procedures Language,"](#) provides the syntax for procedure languages.

Publication Conventions

The following conventions apply to syntax diagrams that appear in this publication.

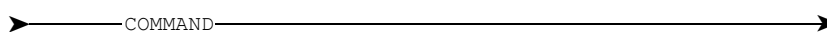
Diagrams are read from left to right along a continuous line (the "main path"). Keywords and variables appear on, above, or below the main path.

Convention	Represents
➤➤	at the beginning of a line indicates the start of a statement.
➤◀	at the end of a line indicates the end of a statement.
————➤	at the end of a line indicates that the statement continues on the line below.
➤————	at the beginning of a line indicates that the statement continues from the line above.

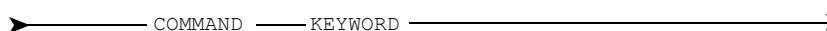
Keywords are in upper-case characters. Keywords and any required punctuation characters or symbols are highlighted. Permitted truncations are not indicated.

Variables are in lower-case characters.

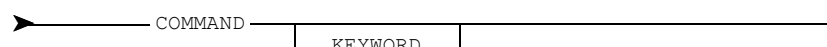
Statement identifiers appear on the main path of the diagram:



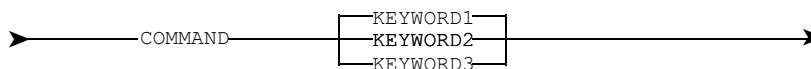
A required keyword appears on the main path:



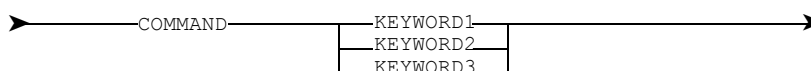
An optional keyword appears below the main path:



Where there is a choice of required keywords, the keywords appear in a vertical list; one of them is on the main path:

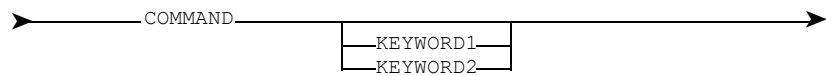


or

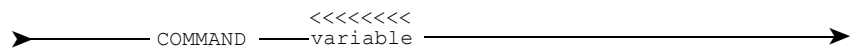


Where there is a choice of optional keywords, the keywords appear in a vertical list, below the main path:

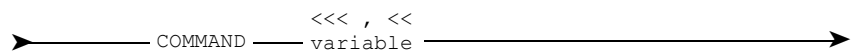
Convention	Represents
\mathbf{A}	Address
\mathbf{B}	Base
\mathbf{C}	Control
\mathbf{D}	Destination
\mathbf{E}	Environment
\mathbf{F}	File
\mathbf{G}	Group
\mathbf{H}	Header
\mathbf{I}	Index
\mathbf{J}	Jump
\mathbf{K}	Key
\mathbf{L}	Label
\mathbf{M}	Memory
\mathbf{N}	Number
\mathbf{O}	Operator
\mathbf{P}	Pointer
\mathbf{Q}	Queue
\mathbf{R}	Register
\mathbf{S}	Stack
\mathbf{T}	Time
\mathbf{U}	Unit
\mathbf{V}	Value
\mathbf{W}	Width
\mathbf{X}	Extension
\mathbf{Y}	Year
\mathbf{Z}	Zone



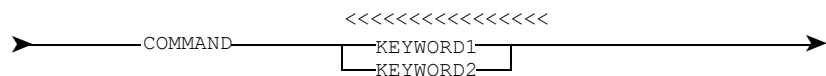
The repeat symbol, <<<<<, above a keyword or variable, or above a whole clause, indicates that the keyword, variable, or clause may be specified more than once:



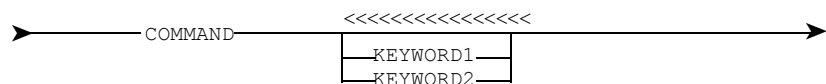
A repeat symbol broken by a comma indicates that if the keyword, variable, or clause is specified more than once, a comma must separate each instance of the keyword, variable, or clause:



The repeat symbol above a list of keywords (one of which appears on the main path) indicates that any one or more of the keywords may be specified; at least one must be specified:



The repeat symbol above a list of keywords (all of which are below the main path) indicates that any one or more of the keywords maybe specified, but they are all optional:



Allen Systems Group, Inc. uses these conventions in publications:

Convention	Represents
ALL CAPITALS	Directory, path, file, dataset, member, database, program, command, and parameter names.
Initial Capitals on Each Word	Window, field, field group, check box, button, panel (or screen), option names, and names of keys. A plus sign (+) is inserted for key combinations (e.g., Alt+Tab).
<i>lowercase italic monospace</i>	Information that you provide according to your particular situation. For example, you would replace <i>filename</i> with the actual name of the file.
Monospace	Characters you must type exactly as they are shown. Code, JCL, file listings, or command/statement syntax. Also used for denoting brief examples in a paragraph.

1

Commands

+ Command

To route to a panel in a different branch of the panel interface and to return to the panel you last saw, after pressing PF3, enter a plus sign in front of an option number.

➤ — + *number* — ➤

where *number* is an option number.

* Line Command

To repeat the previous Line Command against a line.

➤ — * — ➤

/ Line Command

To make any line the Current Line.

➤ — / — ➤

? Command

To:

- Display valid input options. Type ? on the command line.
- Display user-defined help on input fields. Type ? in the input field.

➤ — ? — ➤

??DUP Document Command

To duplicate the following text line a specific number of times when a document frame definition is processed.

```

>--- ??DUP nn ---[ ] ; [ ]

```

where *nn* is an integer.

??EXTRACT Document Command

To insert into an output document, text extracted from specific attributes of a member of any type.

```

>>--- ??EXTRACT MEMBER member-name --- ATTRIBUTE attribute-1 --->
>---[ ] DECOMPOSE ---[ ] YES ---[ ] NO --->
>---[ ] NUMBERING ---[ ] NO ---[ ] YES --->
>---[ ] HEADING attribute-2 ---[ ]
>---[ ] HEADER 'text' ---[ ]
>---[ ] TRANSLATE 'old-char new-char' ---[ ] ; [ ]

```

where:

member-name is the name of a repository member.

attribute-1 is the name of the attribute containing the text to be included in the output document.

attribute-2 is the name of the attribute containing the heading to be included in the output document.

text is a string of text for the heading.

old-char is the character to be translated in the extracted text.

new-char is the character to which *old-char* is to be translated.

??INCLUDE Document Command

To create a reference from one DOCUMENT member to another.

```

>>--- ??INCLUDE MEMBER document-name --->
>---[ ] DECOMPOSE ---[ ] YES ---[ ] NO ---[ ] NUMBERING ---[ ] YES ---[ ] NO --->
>---[ ] HEADER 'text' ---[ ] TRANSLATE 'old-char new-char' ---[ ]
>---[ ] ; [ ]

```


where:

document-name is the name of a DOCUMENT member.

text is a string of text replacing the heading from the TITLE attribute of *document-name*.

old-char is the character to be translated in the CONTENTS attribute of *document-name*.

new-char is the character to which *old-char* is to be translated.

??LINE Document Command

To position a document frame on a specific line of a page.

➤ — ??LINE *number* — [] ; [] — ➤

where *number* is an integer in the range 1 to 68.

??LMAR Document Command

To define the left margin of a document frame definition.

➤ — ??LMAR *right-page left-page* — [] ; [] — ➤

where *right-page* and *left-page* are the margins for the right and left-hand pages.

??PAGE Document Command

To advance printing to a new page.

➤ — ??PAGE — [LEFT — [] ; []
[RIGHT — []
[TO *page-number* — []
[*insert-number* — [] — ➤

where:

page-number is the page number to be skipped to.

insert-number is the number of pages to be inserted.

??PBOT Document Command

To specify the last print line of the page frame definition.

➤—— ??PBOT *right-page left-page* ———— ➤

where *right-page* and *left-page* are the margins for the right and left-hand pages.

??PTOP Document Command

To specify the first print line of the page frame definition.

➤—— ??PTOP *right-page left-page* ———— ➤

where *right-page* and *left-page* are the margins for the right and left-hand pages.

??RMAR Document Command

To define the right margin of a document frame definition.

➤—— ??RMAR *right-page left-page* ———— ➤

where *right-page* and *left-page* are the margins for the right and left-hand pages.

??SKIP Document Command

To insert blank lines in the text of a document.

➤—— ??SKIP ———— *number* ———— *TO line* ———— ➤

where:

number is the number of lines to be inserted, in the range 0 to 67.

line is the line number to be skipped to.

??TITLE Document Command

To create a Table of Contents for a document.

➤—— ??TITLE ———— *NEXT* ———— *DOWN* ———— *UP* ———— *title-text* ———— ➤

where *title-text* is a string of text forming the title.

= Command

To route to a panel in a different branch of the panel interface and to return to the panel you first accessed, after pressing PF3, enter an equals sign in front of an option number.

➤ — = *number* — ➤

where *number* is an option number.

" Line Command

To duplicate a line.

➤ — " — *n* — ➤

where *n* is a number from 1 to 9999.

A Line Command

To:

- Add one or more blank lines to a buffer (Command Interface Environment).

or

- Insert the contents of the Scratchpad *n* times after a line in a buffer (Panel Interface Environment).

➤ — A — *n* — ➤

where *n* is a number from 1 to 9999.

ABBREV Function

To test whether one string is a substring of a second string.

➤ — ABBREV (*string* — *strg* — *length* —) — ➤

where:

string and *strg* are character strings.

length is an integer.

ABEND

This command forces an abend during an ASG-Manager Products (herein called Manager Products) session and allows you the opportunity to produce a dump of virtual storage. This command should only be issued under instructions from the ASG Service Desk.

➤—ABEND—☐;☐—➤

ACCEPT

To save without encoding, all updates made in an assisted update buffer.

➤—ACCEPT—☐;☐—➤

ACT and ACTIVATE

To include a repository member in the project view.

Line Command:

➤—ACT—➤

Primary Command:

➤—ACTIVATE *member-name*—☐;☐—➤

ADD

To insert a member's source record into the repository, generate an encoded record from the newly inserted record, and insert the encoded record into the Data Entries dataset.

➤—ADD *member-name*—☐NUMBER *n*☐;☐—➤
 ➤—*member-definition*—➤
 ➤—☐;☐—➤

where:

member-name is the name of the repository member.

n is an unsigned integer in the range 1 to 10000, being the line number for the first line and the increment for subsequent line numbers of the member's source record.

member-definition is the member's source record.

ADIS and ADISPLAY

To display a repository member in an Assisted Update format. Blank spaces in text attributes are not displayed.

Line Command:

➤ — ADIS — ➤

Primary Command:

➤ — ADISPLAY *member-name* ; — ➤

ADW and IEW EXPORT

To populate ADW or IEW with objects generated from exported repository members.

➤ — ☐ ADW ☐ IEW — EXPORT — *list-name* — ☐ LOCK ☐ NO-LOCK — ➤

➤ — ☐ PRINT ☐ NO-PRINT ☐ NOPRINT — ☐ LSDATE ☐ NO-LSDATE — ☐ CRDATE ☐ NO-CRDATE — ➤

➤ — ☐ UPDATE ☐ NO-UPDATE — ☐ ALIAS ☐ NO-ALIAS — ☐ FILE ☐ NO-FILE — ➤

➤ — ☐ MESSAGE ☐ NO-MESSAGE — ; — ➤

where *list-name* is a KEPT-DATA list name.

ADW and IEW IMPORT

To populate your repository with members generated from imported ADW or IEW objects.

➤ — ☐ ADW ☐ IEW — IMPORT — ☐ RESET ☐ EXTRACT — ➤

➤ — RECONCILE — ☐ NO-COMMON-CLAUSES ☐ COMMON-CLAUSES — ☐ ADDING ☐ REPLACING — ➤

➤ — PREVIEW — *name* — ☐ NEW ☐ APPEND ☐ REPLACE — ☐ PRINT ☐ NO-PRINT ☐ NOPRINT — ➤

➤ — POPULATE — ☐ USER *name* ☐ BUFFER — ☐ ROLLBACK — ☐ PRINT ☐ NO-PRINT ☐ NOPRINT — ➤

➤ — ; — ➤

where *name* is the name of a public USER-MEMBER.

AGET

To insert an assisted update of a member's source record into the current buffer below the current line.

➤➤ — AGET *member-name* — [] ; — ➤➤

AGETC

To replace the contents of the current Update Buffer with an assisted update of the source record of another repository member.

➤➤ AGETC *member-name* _____ ; _____ ➤➤

ALSO KEEP

To add further entries to a KEPT-DATA list, accumulated by an earlier KEEP command.

```

graph LR
    ALSO[ALSO] --> BULK[BULK command]
    BULK --> GLOSSARY[GLOSSARY command]
    GLOSSARY --> LIST[LIST command]
    LIST --> PERFORM[PERFORM command]
    PERFORM --> WHAT[WHAT command]
    WHAT --> WHICH[WHICH command]
    WHICH --> WHOSE[WHOSE command]
    WHOSE --> KEEP[KEEP]
    KEEP --> IN[IN list-name]
    IN --> AND[AND]
    AND --> END[ ]
  
```

where *list-name* is the name of a KEPT-DATA list.

ALTER

To alter the source record of an individual member.

```

ALTER member-name
RENUMBER integer
PRINT
NO-PRINT
PRINT
WIPE-OUT
amendment-line
ACCEPT
QUIT
END
;

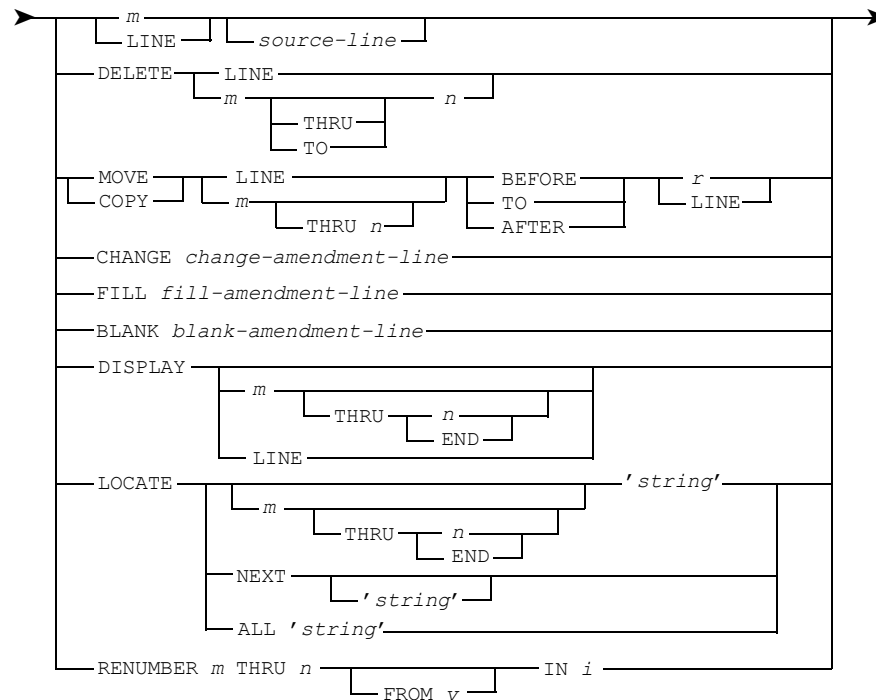
```

where:

member-name is the name of a repository member.

integer is an unsigned integer in the range 1 to 10000.

amendment-line is:



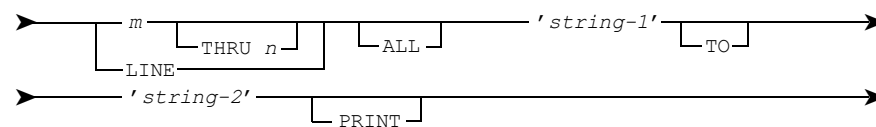
where:

m, *n*, *r*, and *v* are unsigned integers in the range of 1 to 16777215, being line numbers.

i is an integer of increment.

source-line is a member definition statement line appropriate to the member-type.

change-amendment-line is:



where:

m and *n* are as defined above.

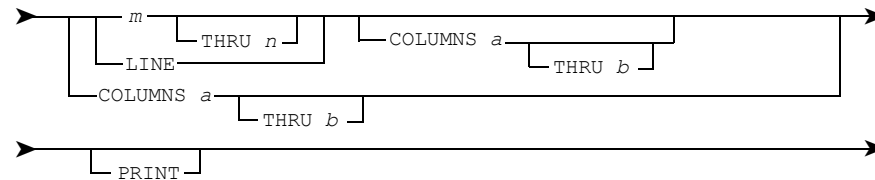
string, *string-1*, and *string-2* are character strings from 1 to 254 characters.

```

graph LR
    In(( )) --> m[m]
    m --> LINE[LINE]
    LINE --> n[THRU n]
    n --> COLUMNS1[COLUMNS a]
    COLUMNS1 --> b1[THRU b]
    b1 --> COLUMNS2[COLUMNS a]
    COLUMNS2 --> b2[THRU b]
    b2 --> string['string']
    string --> PRINT[PRINT]
    PRINT --> Out(( ))
  
```

m , n , and $string$ are as defined above.

blank-amendment-line is:



AMEND

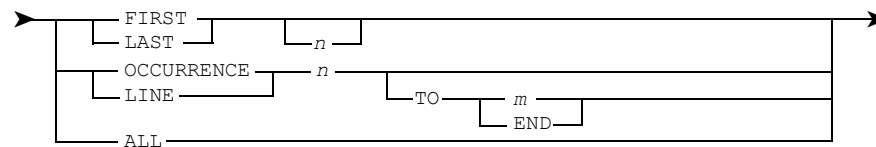
```

AMEND member-name
DELETE attribute-type
REPLACE attribute-type value
ADD attribute-type value
APPEND attribute-type value
CHANGE attribute-type change
DISPLAY
;
.

```

member-name is the name of a member in the repository.

line is:



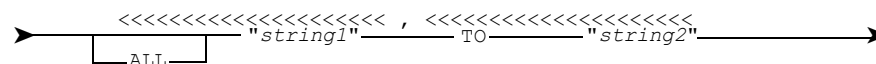
where:

n and m are line numbers of an attribute.

attribute-type is the name of any attribute-type in the member type of the specified member.

value is the value of the attribute.

change is:



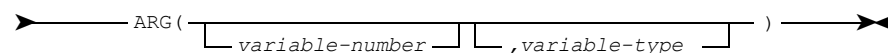
where:

"string1" is the string to be changed.

"*string2*" is the replacement string.

ARG Function

To return the value of an ASG-defined variable.



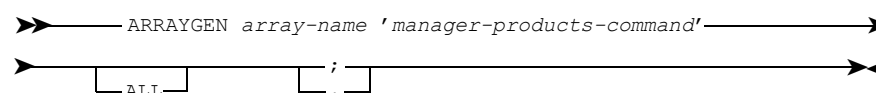
where:

variable-number is an integer in the range 0 to 99.

variable-type is G, I, L, P.

ARRAYGEN Executive Command

To set up an array where each element represents a line of output from a specified Manager Products command.



where:

array-name is the name of an array to store the output from the Manager Products command.

manager-products-command is a Manager Products command input string, optionally including a terminator.

ARRAYHI Function

To return the number of the highest used element of an array.

➤ — ARRAYHI (*expression*) — ➤

where *expression* evaluates to a user-defined variable name.

ARRAYLO Function

To return the number of the lowest used element of an array.

➤ — ARRAYLO (*expression*) — ➤

where *expression* evaluates to a user-defined variable name.

ARRAYSORT Procedures Language Executive Command

To sort the contents of a command or global array.

➤ — ARRAYSORT — *source-array-name* — COMMAND — ➤
GLOBAL —
 ➤ — *target-array-name* — COMMAND — ➤
GLOBAL —
 ➤ — *key-start* — *key-length* — ASCENDING — ➤
DESCENDING —
 ➤ — ; — ➤

where:

key-start must be numeric within the range 1 to 255.

key-length must be numeric within the range 1 to 255.

ASM

To generate an ASSEMBLER data structure.

Primary Command:

```
➤——ASM member-name——[ ] ; [ ]——➤
```

where *member-name* is the name of a repository member.

Line Command:

```
➤——ASM——➤
```

AUPDATE and AUPD

To update the source record of a repository member together with a skeleton to assist in the entry of data into the buffer.

Primary Command:

```
➤——AUPDATE member-name member-type——[ ] ; [ ]——➤
```

where:

member-name is the name of the member being created or updated. Member names must have a prefix valid for your repository.

member-type is the name of a member-type for which a member is to be updated. Member-type must be specified in environments that use no naming convention. If member-type name is not specified, assisted update will not know which skeleton to set up for the member being updated.

Line Command:

```
➤——AUPD——➤
```

AUTHORITY

To identify yourself as the user requiring access to the repository.

```
➤——[ AUTHORITY ]——password——[ ] ; [ ]——➤
```

where *password* is a character string of up to eight printable or non-printable characters, being the identification issued by the Controller to the particular user or to the Master Operator. This must be delimited if it contains characters from the extended character set.

B Line Command

To insert n copies of the Scratchpad before the line containing the B (Panel Interface Environment).

```

>> — B — [ n ] —————><

```

where n is an integer in the range 1 to 9999.

BACKWARD

To display the preceding screen in the current buffer.

```

>> — BACKWARD [ ; ] —————><

```

BOTTOM

To display the last screen in the current buffer.

```

>> — BOTTOM [ ; ] —————><

```

BROWSE and LOOKASIDE

To display output in a new Lookaside buffer.

```

>> — [ BROWSE ] — command — [ ; ] —————><
    [ LOOKASIDE ]

```

where *command* is the name of a command.

BS2000 HELP

To obtain information about the latest changes to the Siemens version of Manager Products.

```

>> — BS2000 — [ HELP ] ————— [ ; ] —————><
               [ H ]

```

BUILD Executive Command

To allow the exchange of data between a KEPT-DATA list and an array for subsequent processing either within an Executive Routine or in the Command Area.

```

>> — BUILD — [ build-kept-data-from-array-clause ] [ ; ] —————><
               [ build-array-from-kept-data-clause ] [ . ]

```

where *build-kept-data-from-array-clause* is:

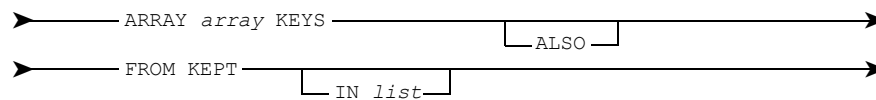


where:

list is the name of a KEPT-DATA list.

array is the name of an array containing key values.

build-array-from-kept-data-clause is:



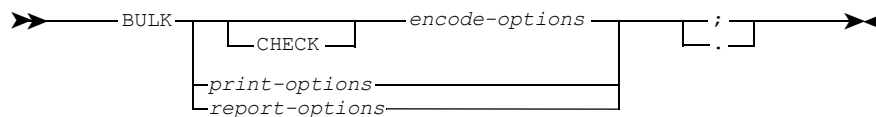
where:

array is as defined above.

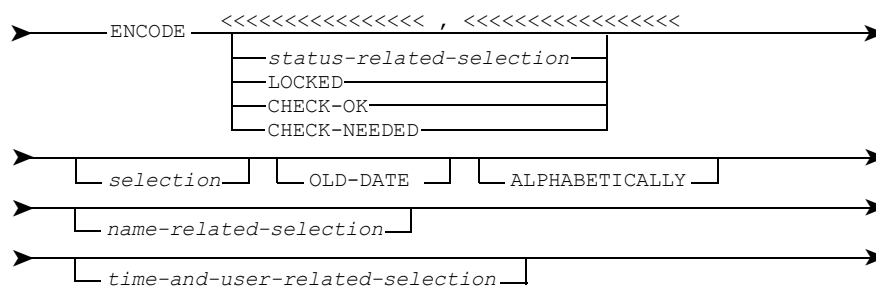
$list$ is as defined above.

BULK ENCODE, BULK PRINT, and BULK REPORT

To apply an ENCODE, a PRINT, or a REPORT command to all members or to selected members of the repository.



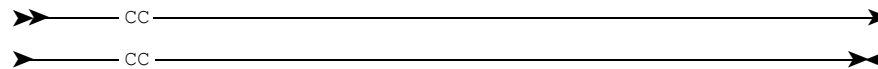
where *encode-options* is:



Note:

The keyword OLD-DATE is only available to the Controller.

where n is an integer in the range 1 to 9999.



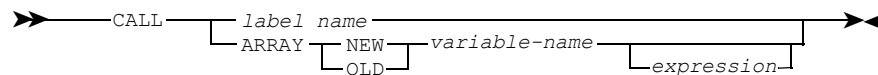
CALCULATE

To carry out load factor calculations for every data element referenced in any USERVIEW in the Workbench Design Area.



CALL Directive

To cause an immediate branch to a label or to execute a user-defined variable array as a procedure.



where:

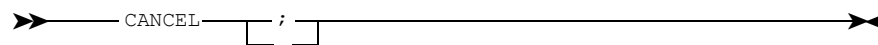
label-name is the name of a label.

variable-name is the name of a user-defined variable array which may have no more than ten characters.

expression is a parameter to be passed when the user-defined variable array is executed as a procedure.

CANCEL

To cancel all updates made in an assisted update buffer.



CC Line Command

See ["C and CC Line Commands" on page 17](#).

CENTER Function

To return a string centered in a field of specified length.

where:

string is a string.

length is an integer.

pad is a character.

CEXEC

See ["CORPORATE-EXECUTIVE or CEXEC" on page 27](#).

CHANGE

To change one character string to another or to delete a character string.

where:

/ is the delimiter character.

old is the character string to be changed or deleted.

new is the character string to be inserted.

mm is the number of lines (starting from the Current Line) on which the string is to be changed or deleted.

* denotes all.

nn is the number of occurrences of the string which are to be changed or deleted.

CHECK

To perform integrity checks as members are encoded or removed.

where *command* is a command that encodes or removes members, except AUPD, AUPDATE, DCUPD, PUR, or PURGE.

CLEAR

To clear the contents of the buffer displayed on the screen.

```
➤—— CLEAR ——— [ ] ; [ ] ——➤
```

CLOSE

To close a dataset external to Manager Products or to close a repository.

```
➤—— CLOSE ——— [ ] ; [ ] ——➤
      |
      |—— DICTIONARY ——— [ ]
      |
      |—— DATASET ——— data-set-name ——— [ ]
      |
      |—— FILE ——— [ ]
      |
      |—— EXTERNAL ——— [ ]
```

where *data-set-name* is the logical file name (ddname or dtfname) as declared in job control statements and as used to indicate the name of the external dataset (physical file) you wish to CLOSE. The name can be up to eight characters long for OS, CMS, or BS2000 environments, or seven characters for DOS.

CLOSEF Executive Command

To close a particular output destination or all destinations defined in previous SENDF commands.

```
➤—— CLOSEF — options ——— [ ] ; [ ] ——➤
```

where *options* is:

```
➤—— [ USER-MEMBER ——— mp-user-name ——— ]
      |
      |—— [ PUBLIC-USER-MEMBER ——— ]
      |
      |—— [ PRIVATE-USER-MEMBER ——— ]
      |
      |—— [ SEQUENTIAL file-name ——— ]
      |
      |—— [ PRINT ——— ]
      |
      |—— [ ALL ——— ]
```

where:

mp-user-name is the name of a destination MP-AID USER-MEMBER.

file-name is the name of a sequential dataset. It is the logical file name (ddname or dtfname) used in job control statements to define the external dataset name (physical file name) of the file.

CMS

To enable a machine to switch from a Manager Products environment to a virtual machine environment.

COB and COBOL

To generate a COBOL data structure.

Line Command:

A horizontal line with arrows at both ends. The label "COB" is centered above the line.

Primary Command:

Diagram illustrating the syntax for the `COBOL member-name` field. The field is defined as `COBOL member-name ;`, where the semicolon is a required terminator.

where *member-name* is the name of a repository member.

COMMAND Directive

To declare Command Variables.

COMMAND <<<<<<<<<<<<<<< user-variable-name

where *user-variable-name* is any valid user-defined variable name.

CONTROL NEW-ALIASES

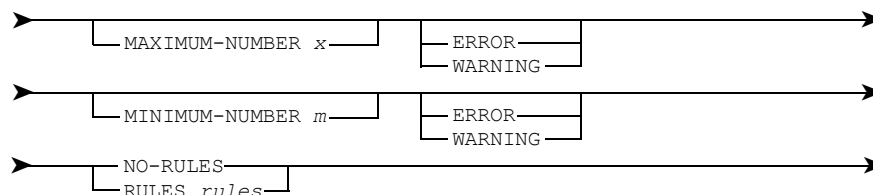
To alter the alias table used to control the encoding of the ALIAS clauses of data definition statements, to conform to your installation's particular requirements.

```

➤-----CONTROL NEW-ALIASES-----➤
➤-----ALL-ALIASES all-rules-----➤
|-----GENERAL-ALIASES general-rules-----|
|-----REMOVE alias-type-----|
|-----REPLACE alias-type-----|
|-----ADD alias-type-----|
|-----SHOW-ALIASES-----|
|-----QUIT-----|
|-----;-----|
|-----.-----|

```

where *all-rules* is:

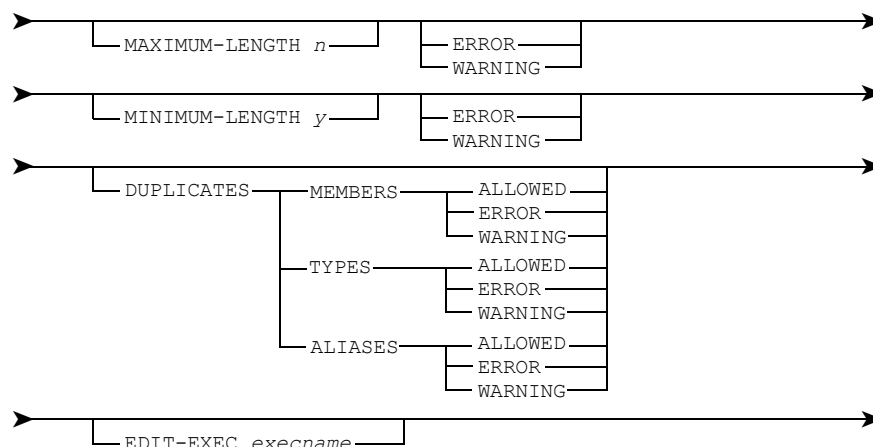


where:

x is an integer in the range 1 to 254.

m is an integer in the range 1 to 254. *m* must be greater than or equal to *x*, if used.

rules is:



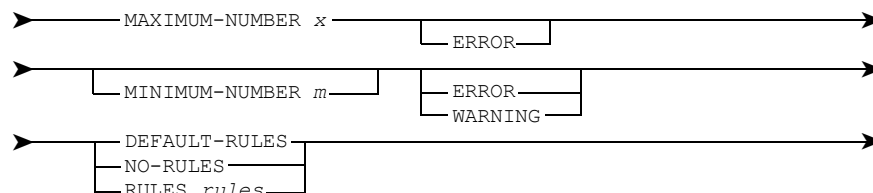
where:

n is an integer in the range 1 to 79.

y is an integer in the range 1 to 79. y must be greater than or equal to n , if used.

execname is the name of a user-defined Executive Routine and may be up to 10 characters in length.

general-rules is:

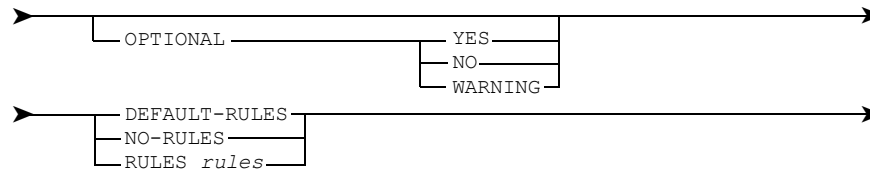


where:

x , m , and $rules$ are as defined above.

$alias-type$ is the name of a specific alias-type.

$specific\ rules$ is:



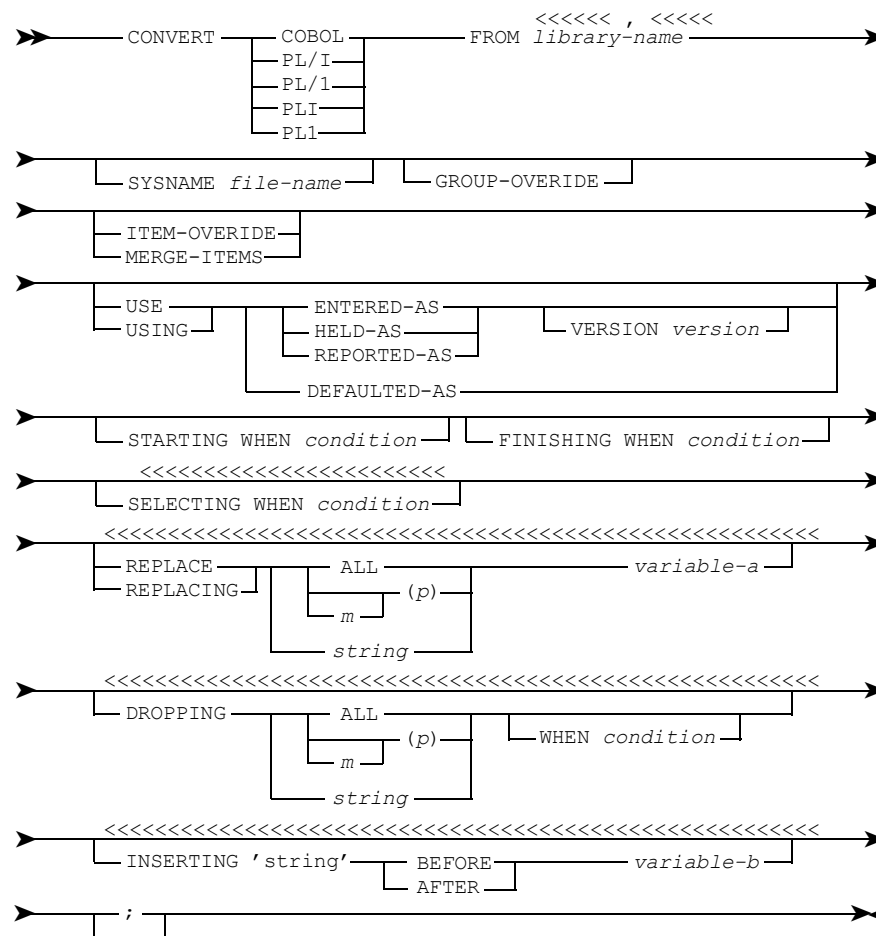
where:

$rules$ is as defined above.

$synonym$ is the name of an alias-type synonym.

CONVERT

To insert into the repository member source records generated from COBOL or PL/I source data descriptions.



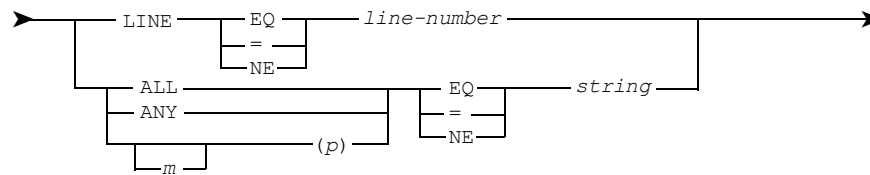
where:

library-name is the name to be given to the generated library member in the output dataset. The name must not be more than eight characters long. The first character must be alphabetic or #, £ (or local currency symbol with the internal code hexadecimal 5B), %, or !.

file-name is the logical file name of the dataset that contains the selected source members or source copybooks.

version is an unsigned integer of up to 2 digits in the range 1 to 15.

condition is:



where:

line-number is an unsigned integer of up to eight digits.

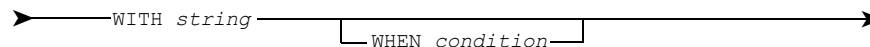
EQ or = specifies equal to.

NE specifies not equal to and cannot be used in a STARTING clause.

m and (*p*) specify part of a source data name beginning at character position *m* and containing *p* characters. If *m* is omitted then a value of 1 is assumed. (*p*) must always appear in parentheses. The sum of both integers must not exceed 97. If *m* and (*p*) are both present, they must be separated by at least one space.

string is any printable character string, up to 32 characters in length.

variable-a is:

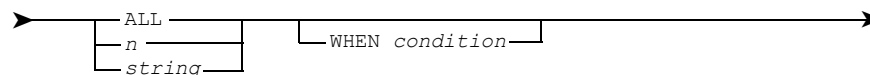


where:

string is as defined above.

condition is as defined above.

variable-b is:



where:

n is an unsigned integer in the range 1 to 96, specifying a character position in a generated data name condition is as defined above.

string is as defined above.

COPIES Function

To return concatenated copies of a specified string.

➤——COPIES (*string*,*number*)——➤

where:

string is a string.

number is the total of concatenated copies of string.

COPY

To copy the source record of a member and either:

- Insert that copy as the source record of a new member.

or

- Insert that copy as the source record of an existing member (replacing the previous source record).

➤——COPY *member-name-1*——➤

➤——TO *member-name-2*——➤

 └──FROM *status-name*──┘

 └──FROM *status-name*──┘

 └──TO *member-name-2*──┘

➤——[ENCODED-SOURCE] [REPLACE] [;]——➤

where:

member-name-1 is the name of the existing repository member whose source record is to be copied.

member-name-2 is the name of the new repository member.

status-name is the name of a status.

COPYRIGHT

To output a legal Copyright statement concerning ASG and other Vendor ownership of ASG Program Products and/or Derived Programs.

➤——COPYRIGHT——[;]——➤

CORPORATE-EXECUTIVE or CEXEC

To execute a Corporate Executive Routine.

where *name* is the name of a Corporate Executive Routine.

D and DD Line Commands

To delete one or more lines.

A horizontal line representing a metric space D . The line starts with a double arrow pointing left and ends with a double arrow pointing right. A bracket below the line indicates a subset of size n .

where n is an integer in the range 1 to 9999.

DD

DD

DACCESS

To bring information about an index-name into virtual storage for further processing.

The diagram illustrates the structure of the SQL statement `DESCRIBE`. It is divided into three main sections:

- Access and Condition:** The first section starts with `DACCESS` and is followed by an optional `CONDITION`.
- Member and Catalogue:** The second section lists the `MEMBER member-name` and the `CATALOGUE classification`.
- Alias and Attribute:** The third section lists the `ALIAS alias-name` and the `ATTRIBUTE attribute-name`.
- Key Variable:** The fourth section lists the `KEY key-variable-name`.
- Cursor and With:** The fifth section starts with `CURSOR c` and is followed by an optional `WITH`.
- Situation and Details:** The sixth section lists the `SITUATION`, `USED-BY`, `REFERENCES`, and `STATUS-DETAILS`.
- Suppress and Attributes:** The seventh section starts with `SUPPRESS` and is followed by an optional `ATTRIBUTES`.
- Counts and Definition:** The eighth section lists the `COUNTS`, `ALL-ATTRIBUTES`, and `DEFINITION`.

where:

member-name is the name of a repository member.

classification is a catalog classification.

alias-name is the name of an alias.

attribute-name is the name of an indexed attribute.

key-variable-name is the name of a numeric key identifying a member, alias, catalog, or indexed attribute on the Manager Products repository.

c is an integer in the range 1 to 32767.

DACT and DEACTIVATE

To remove a repository member from the project view.

➡➡ — DACT ————— ➡

Primary Command:

➤➤ — DEACTIVATE *member-name* — ; — ➤➤

DB2 ALTER

DB2 ALTER generates one or more SQL ALTER TABLE, INDEX, TBSPACE, STOGROUP, FUNCTION, PROCEDURE, TRIGGER, or TYPE statements from the definition of a DB2-TABLE, DB2-INDEX, DB2-TBSPACE, DB2-STOGROUP, DB2-PROCEDURE, DB2-TRIGGER, or ITEM member.

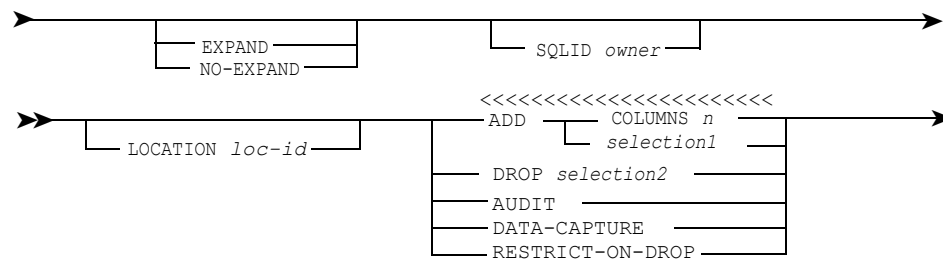
The diagram illustrates the syntax of the DB2 ALTER command. It shows a sequence of components: DB2 ALTER, table-name, table-options, index options-1, CONVERT-TYPE, tbspace, options-1, LOCKSIZE, LOCKMAX, COMPRESS, stogroup, stogroup-options, database, database-options, USING user-exit, name-editing options, ONTO destination, PRINT, NOPRINT, and a semicolon (;). The components are connected by lines indicating their relative positions and groupings within the command structure.

where:

table, *index*, *tblspace*, *stogroup*, and *database* are names of DB2-TABLE, DB2-INDEX, DB2-TBSPACE, DB2-STOGROUP, and DB2-DATABASE members respectively.

user-exit is the name of a user-created executive routine.

table-options are:



Note:

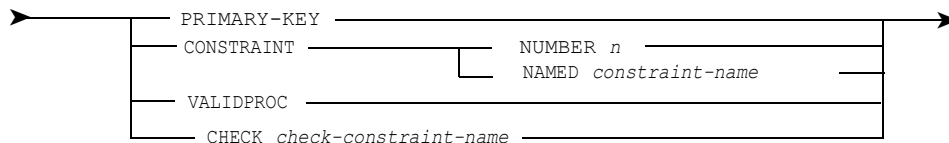
The **AUDIT** keyword can only be specified once, and must immediately precede or follow all **ADD/DROP** selections.

where:

owner is a delimited string of up to 8 characters, giving the authorization ID of a particular user

loc-id is a delimited string of up to 16 characters, giving the name of a DB2 location.

selection1 is:



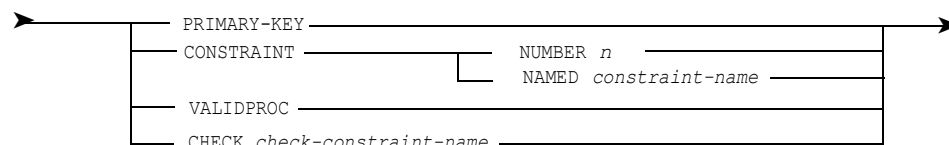
where:

n is an integer

constraint-name is a name specified in the NAMED attribute of a DB2-TABLE member

check-constraint-name is an SQL long identifier having up to 18 characters.

selection2 is:

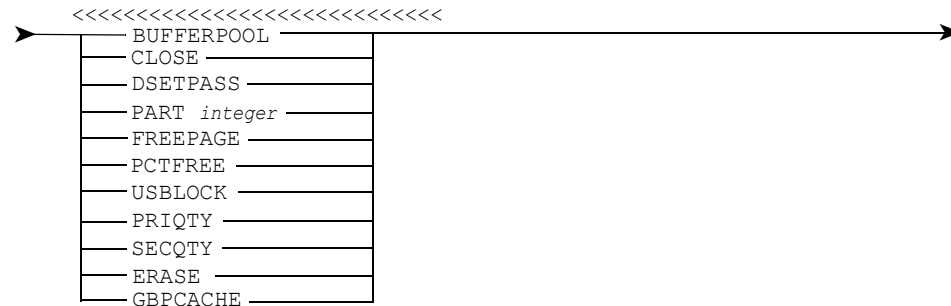


where:

constraint-name is an SQL long identifier having up to 18 characters

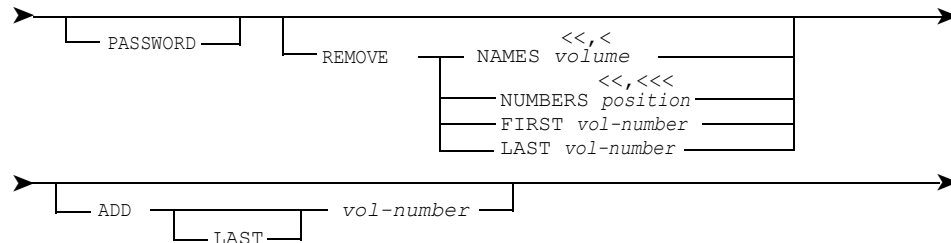
check-constraint-name is an SQL long identifier having up to 18 characters.

options-1 are:



where *integer* is a partition number.

stogroup-options are:



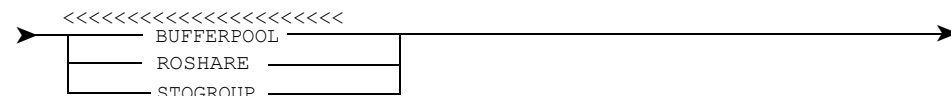
where:

vol-number is an integer giving a number of volumes

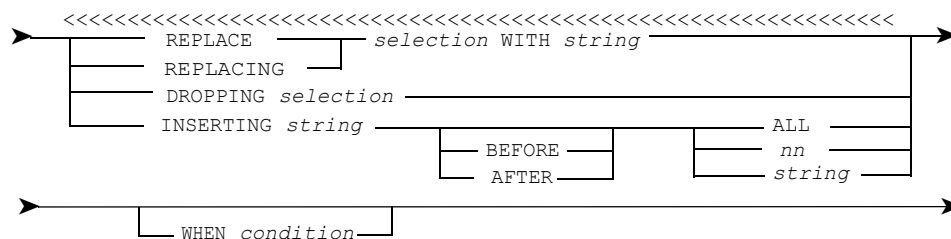
volume is a delimited string giving a volume name

position is the position of a volume in the list of volumes in the DB2-STOGROUP member definition.

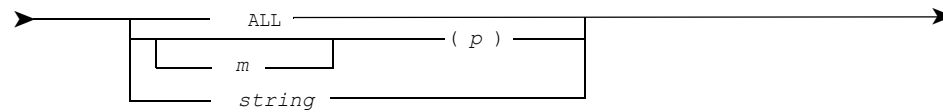
database-options are:



name-editing-options are:



where *selection* is:



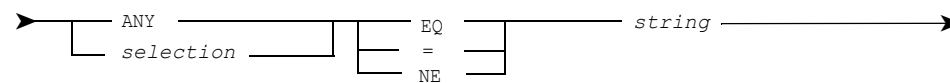
where:

m and *p* are integers in the range 1 to 96

string is a delimited string of not more than 32 printable characters

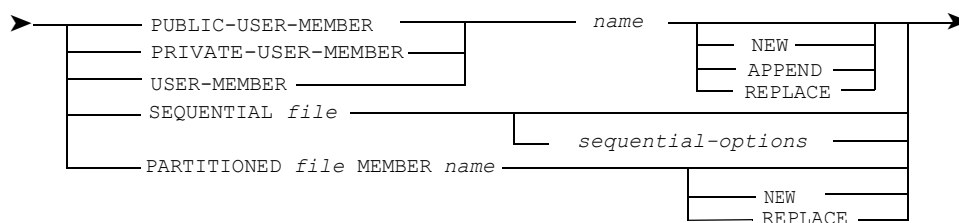
nn is an unsigned integer in the range 1 to 96.

condition is:



where *selection* and *string* are as defined above.

destination is:

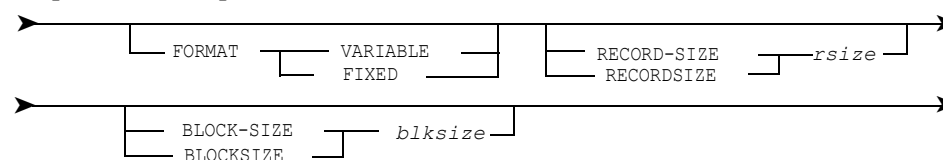


where:

name is the name of a USER-MEMBER

file is the name of a sequential or partitioned dataset.

sequential-options are:



where:

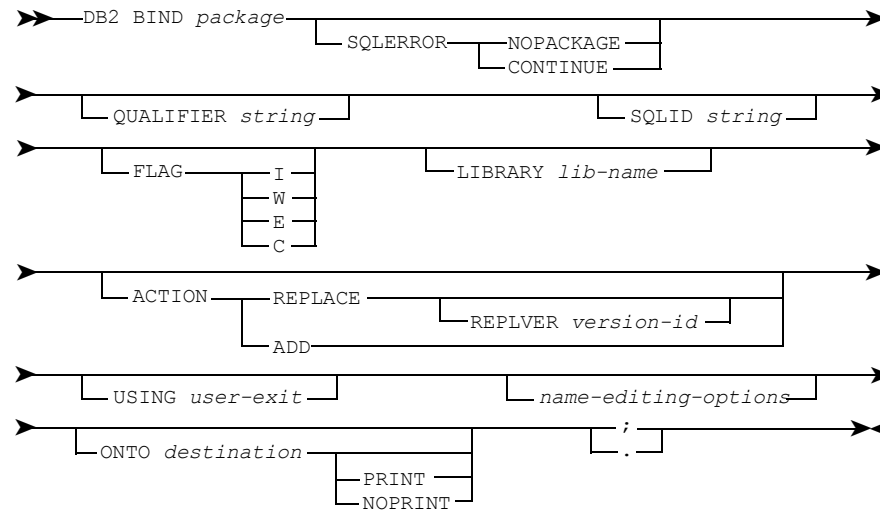
rsiz is the record length

blksize is the block size.

DB2 BIND and REBIND

To generate BIND or REBIND subcommands for your DB2 environment.

Syntax of DB2 BIND PACKAGE



where:

package is the name of a DB2-PACKAGE repository member.

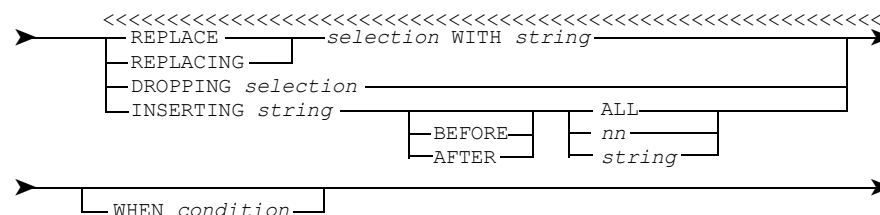
string is a string of 1 to 8 characters, delimited.

lib-name is the name, delimited, of a partitioned dataset that contains the DBRM that is being bound into the package.

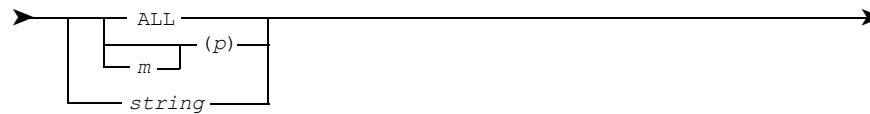
version-id is a string 1 to 64 characters, delimited.

user-exit is a delimited string of up to 16 characters, giving the name of an Executive Routine.

name-editing-options are:



where *selection* is:



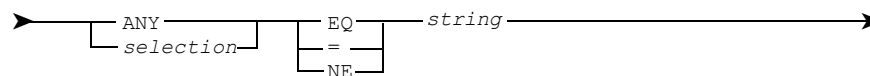
where:

m and *p* are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

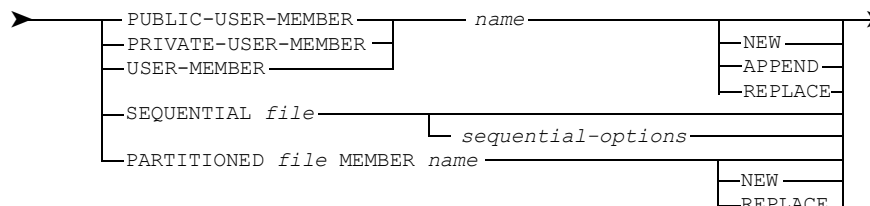
condition is:



where:

selection and *string* are as defined above.

destination is:

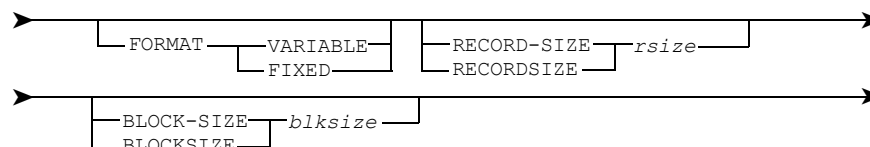


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:

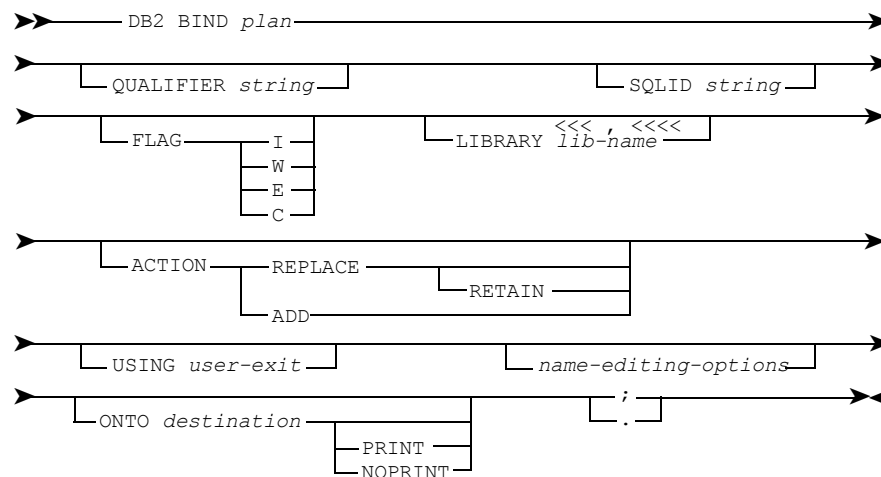


where:

*rsiz*e is the record length.

blksize is the block size.

Syntax of DB2 BIND PLAN



where:

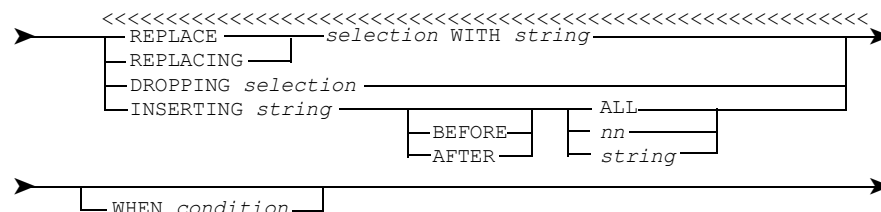
plan is the name of a DB2-PLAN repository member.

string is a string of 1 to 8 characters, delimited.

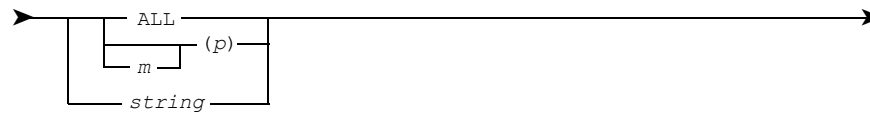
lib-name is the delimited name of 1 or more partitioned datasets (libraries) that contain as members the DBRMs specified.

user-exit is a delimited string of up to 16 characters, giving the name of an Executive Routine.

name-editing-options are:



where *selection* is:



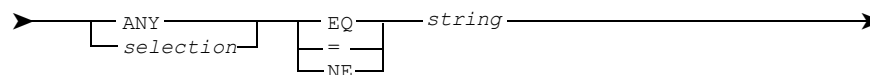
where:

m and *p* are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

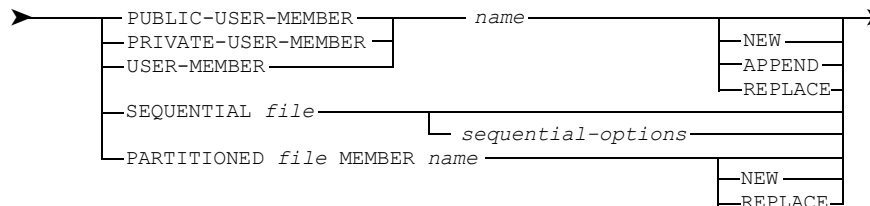
condition is:



where:

selection and *string* are as defined above.

destination is:

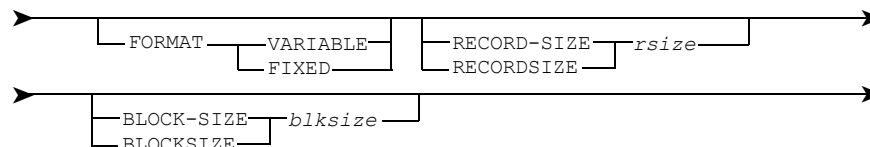


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:

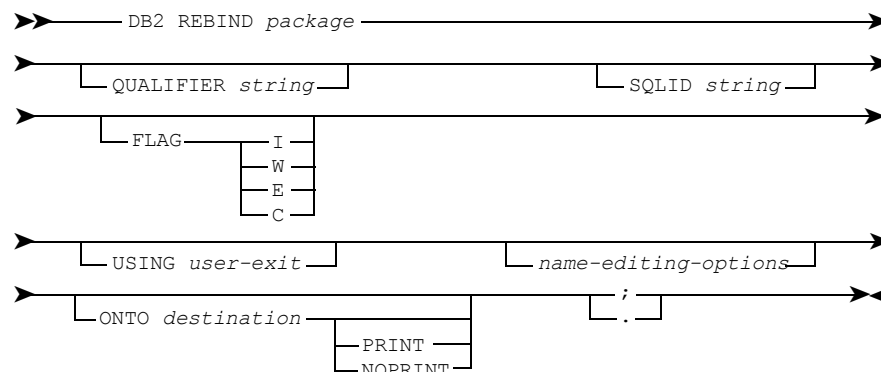


where:

*rsiz*e is the record length.

blksize is the block size.

Syntax of DB2 REBIND PACKAGE



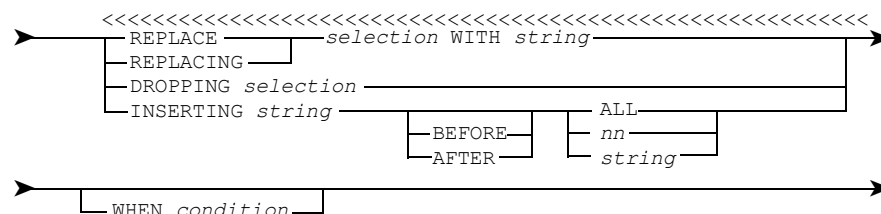
where:

package is the name of a DB2-PACKAGE repository member.

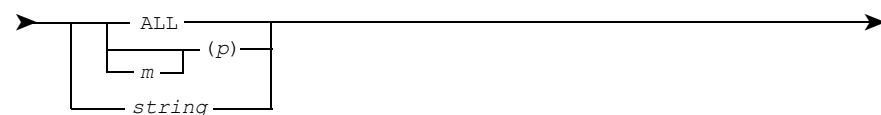
string is a string of 1 to 8 characters, delimited.

user-exit is a delimited string of up to 16 characters, giving the name of an Executive Routine.

name-editing-options are:



where *selection* is:



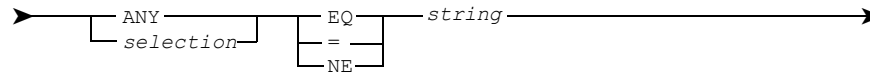
where:

m and p are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

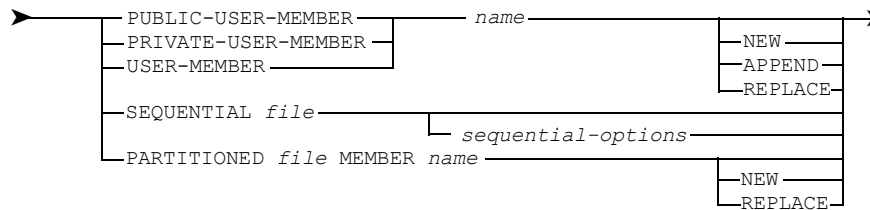
condition is:



where:

selection and *string* are as defined above.

destination is:

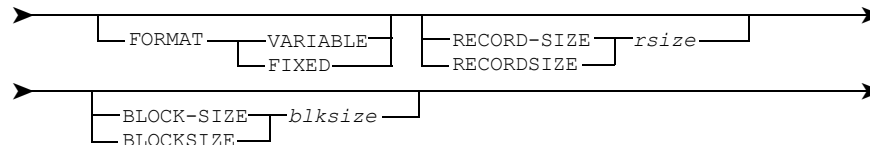


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:

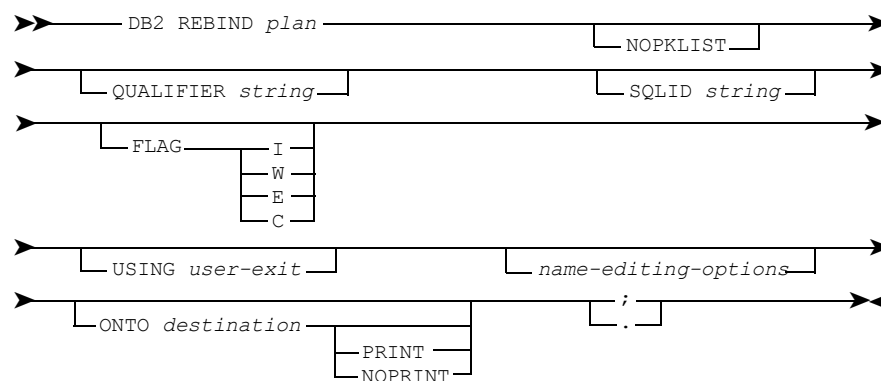


where:

rsizelength is the record length.

blksize is the block size.

Syntax of DB2 REBIND PLAN



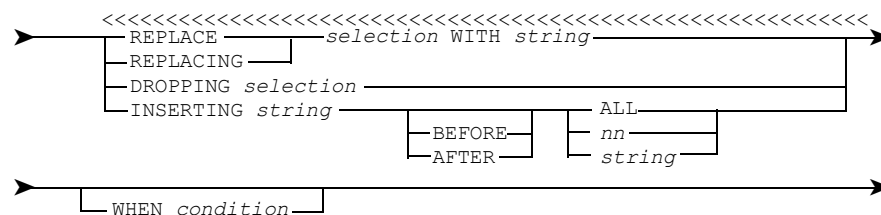
where:

plan is the name of a DB2-PLAN repository member.

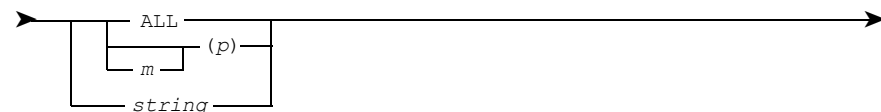
string is a string of 1 to 8 characters, delimited.

user-exit is a delimited string of up to 16 characters, giving the name of an Executive Routine.

name-editing-options are:



where *selection* is:



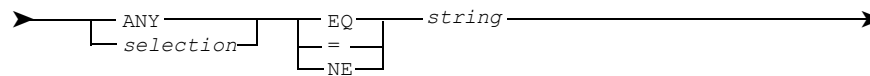
where:

m and p are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

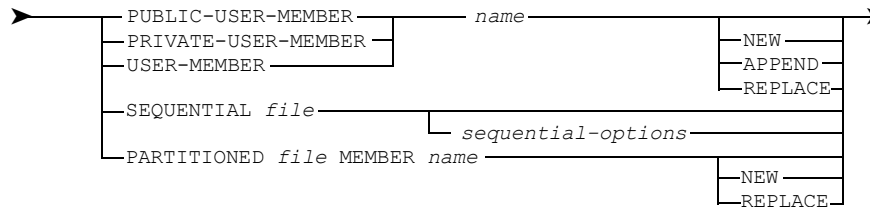
condition is:



where:

selection and *string* are as defined above.

destination is:

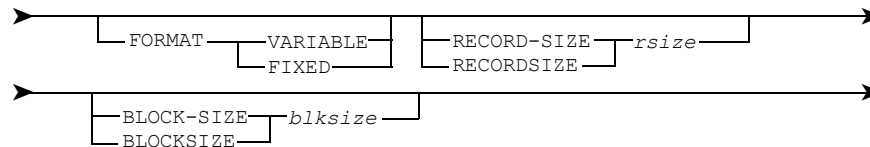


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



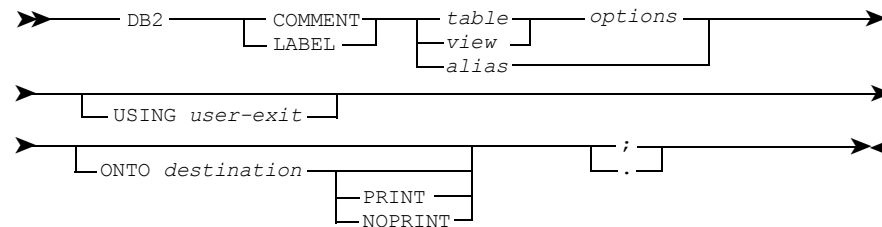
where:

rsize is the record length.

blksize is the block size.

DB2 COMMENT and DB2 LABEL

To generate SQL COMMENT ON or LABEL ON statements from the definition of a DB2-TABLE, DB2-VIEW, or DB2-ALIAS member.



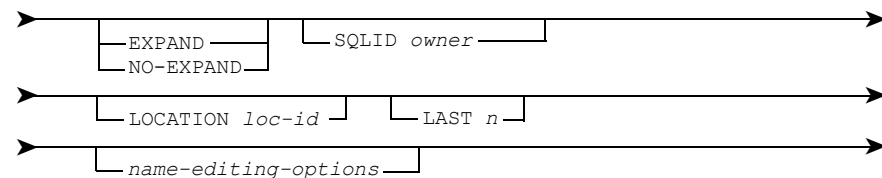
where:

table is the name of a DB2-TABLE member.

view is the name of a DB2-VIEW member.

alias is the name of a DB2-ALIAS member.

options are:



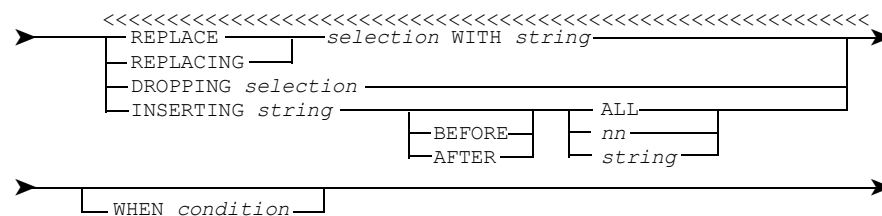
where:

owner is a delimited string of up to 8 characters, giving the authorization ID of a specific user.

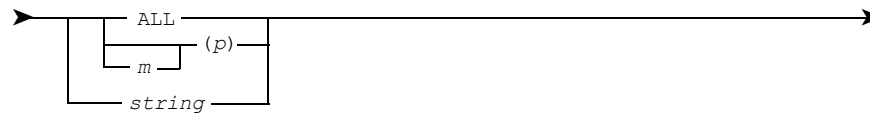
loc-id is a delimited string of up to 16 characters, giving the name of a DB2 location.

n is an integer.

name-editing-options are:



where *selection* is:



where:

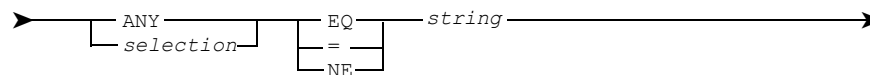
m and *p* are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

string is as defined above.

condition is:

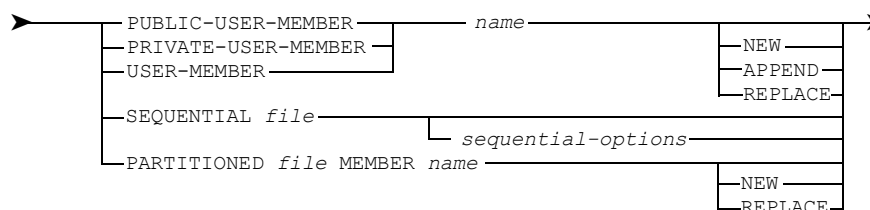


where:

selection and *string* are as defined above.

user-exit is the name of an Executive Routine.

destination is:

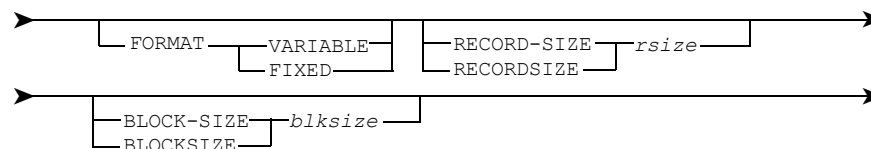


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



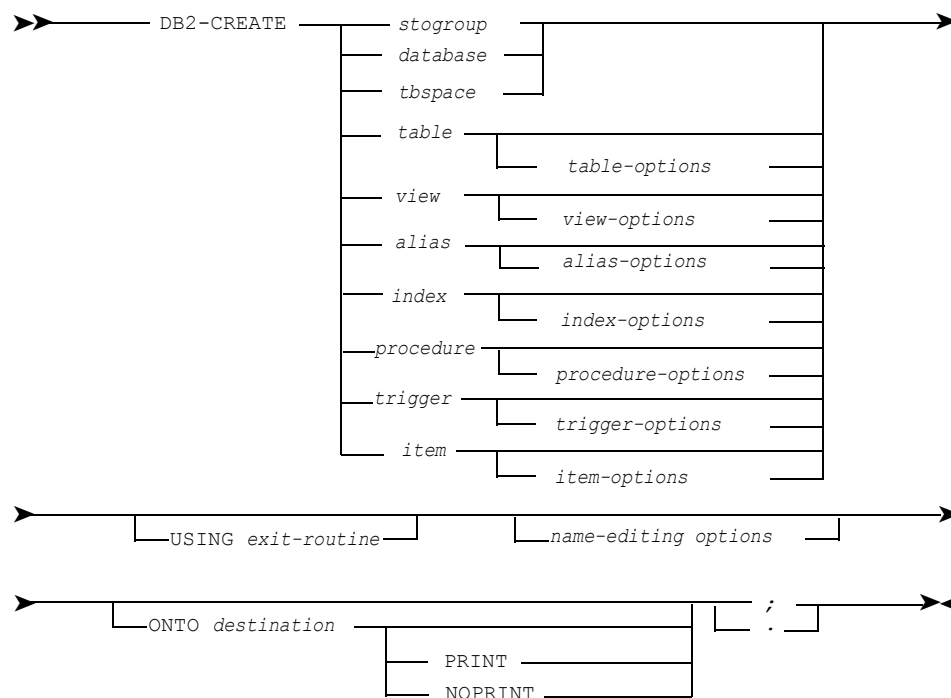
where:

rsize is the record length.

blksize is the block size.

DB2 CREATE

To generate an SQL CREATE statement for a DB2 object from its definition in a repository member.

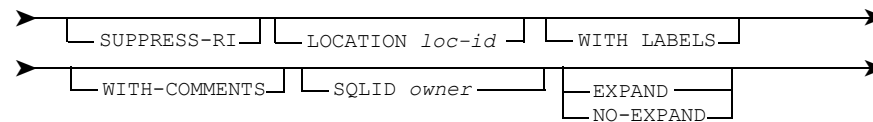


where:

alias, *database*, *index*, *stogroup*, *table*, *tablespace*, *procedure*, *trigger*, *item*, and *view* are names of DB2-ALIAS, DB2-DATABASE, DB2-INDEX, DB2-STOGROUP, DB2-TABLE, DB2-TBSPACE, and DB2-VIEW repository members.

exit-routine is the name of an Executive Routine.

table-options are:

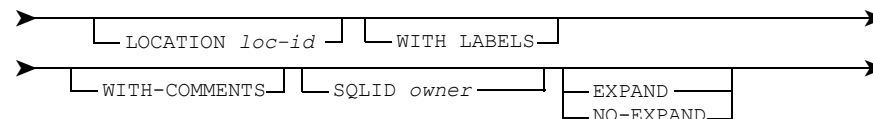


where:

loc-id is a delimited string of up to 16 characters, giving the name of a DB2 location.

owner is a delimited string of up to 5 characters, giving the ID of a specific user.

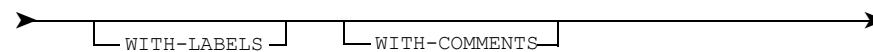
view-options are:



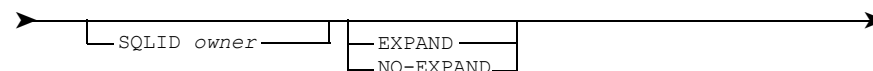
where:

loc-id and *owner* are as defined above.

alias-options are:



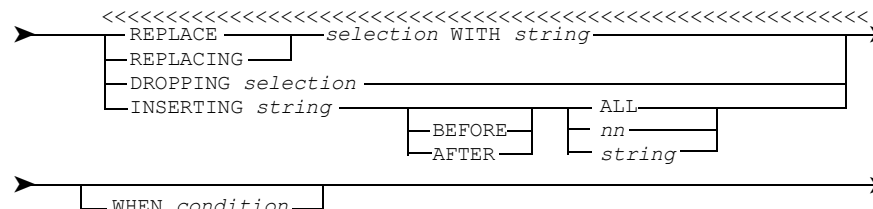
index-options are:



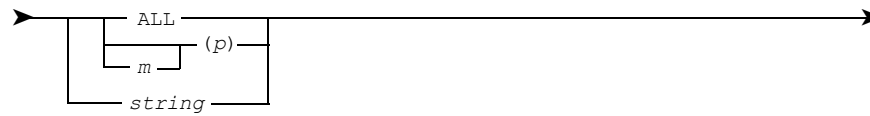
where:

owner is as defined above.

name-editing-options are:



where *selection* is:



where:

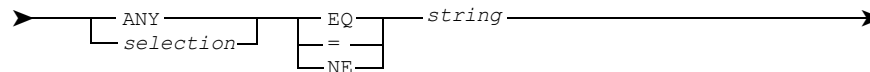
m and *p* are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

string is as defined above.

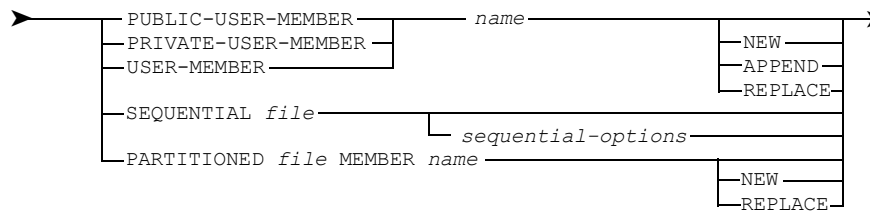
condition is:



where:

selection and *string* are as defined above.

destination is:

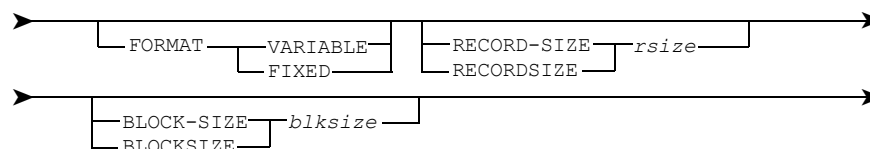


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



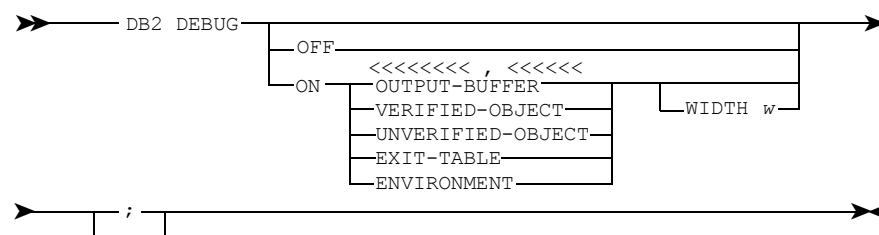
where:

*rsiz*e is the record length.

blksize is the block size.

DB2 DEBUG

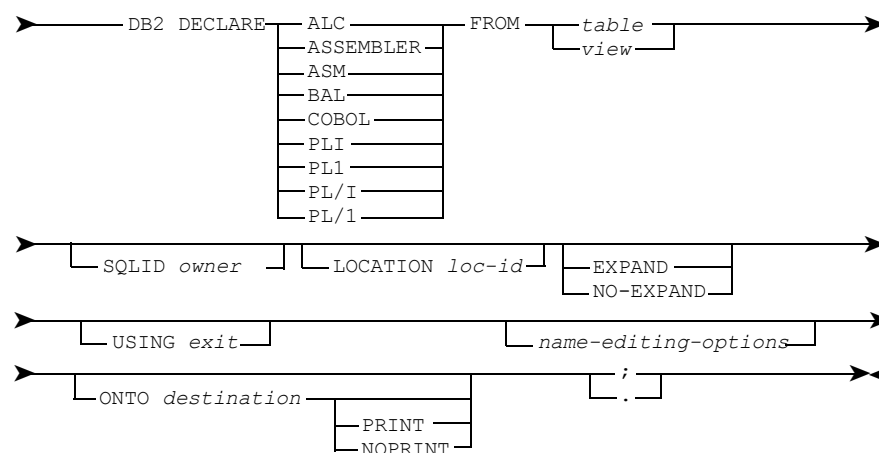
To produce diagnostic information during DB2 export commands.



where *w* is the display width.

DB2 DECLARE

To generate an SQL DECLARE TABLE statement in COBOL, PL1, or Assembler language, from a DB2-TABLE or DB2-VIEW repository member.



where:

table is the name of a DB2-TABLE repository member.

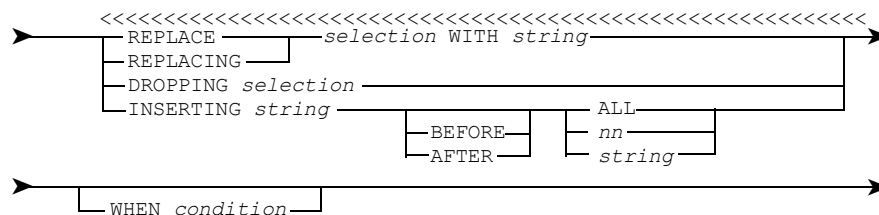
view is the name of a DB2-VIEW repository member.

owner is a delimited string of up to 8 characters, giving the authorization ID of a particular user.

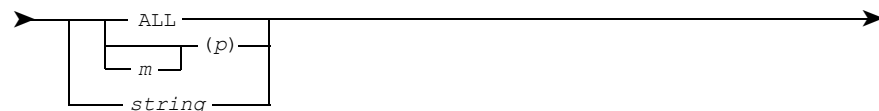
loc-id is a delimited string of up to 16 characters, giving the name of a DB2 location.

exit is the name of an user-created Executive Routine.

name-editing-options are:



where *selection* is:



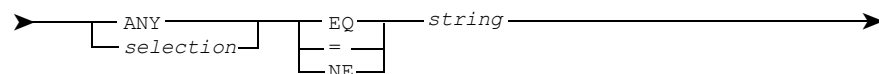
where:

m and p are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

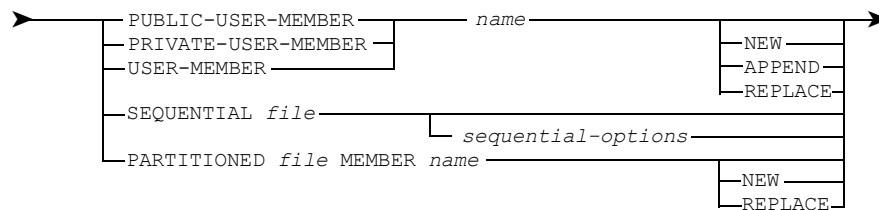
condition is:



where:

selection and *string* are as defined above.

destination is:

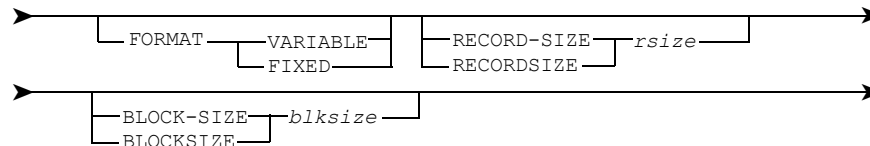


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



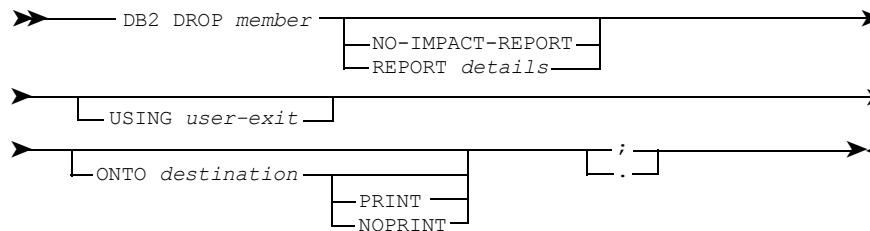
where:

rsize is the record length.

blksize is the block size.

DB2 DROP

To generate an SQL DROP statement and an optional impact analysis report.



where:

member is the name of a DB2-ALIAS, DB2-DATABASE, DB2-INDEX, DB2-STOGROUP, DB2-TABLE, DB2-TBSPACE, DB2-PROCEDURE, DB2-TRIGGER, ITEM, or DB2-VIEW repository member.

details are:



where:

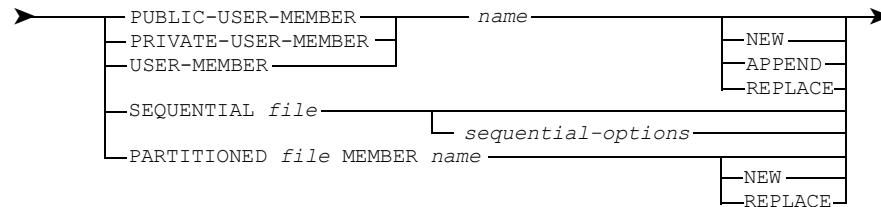
w is an integer between 50 and 246.

i is an integer between 5 and 20.

k is the name of a KEPT-DATA list.

user-exit is the name of an Executive Routine.

destination is:

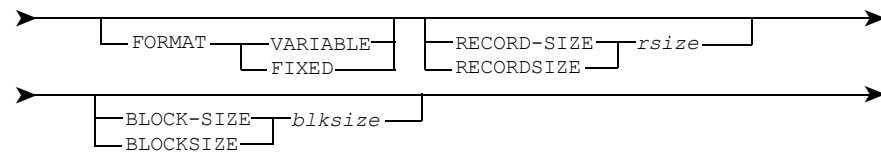


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



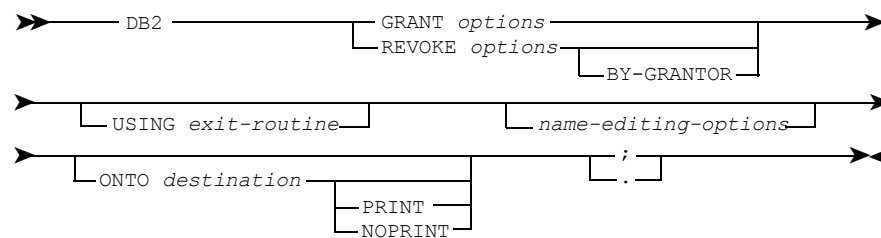
where:

rsize is the record length.

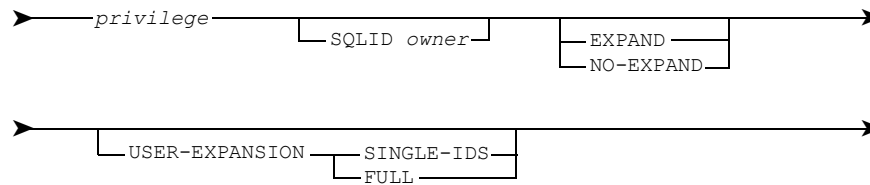
blksize is the block size.

DB2 GRANT and DB2 REVOKE

To generate an SQL GRANT or REVOKE statement for a DB2 object from its definition in a DB2-PRIVILEGE repository member.



where *options* are:



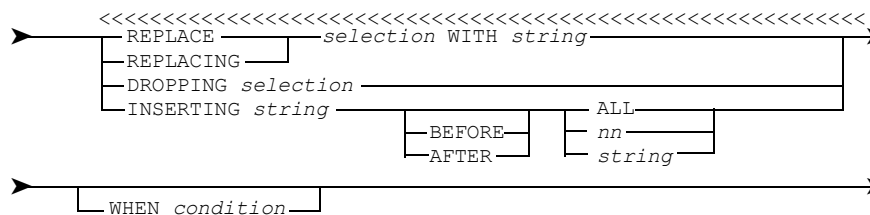
where:

privilege is the name of a DB2-PRIVILEGE member.

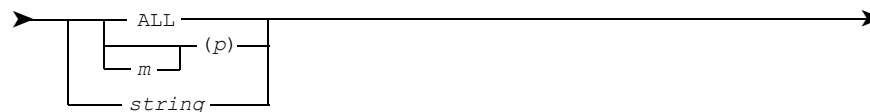
owner is a delimited string of up to eight characters, giving the authorization ID of a particular user.

exit-routine is the name of an Executive Routine.

name-editing-options are:



where *selection* is:



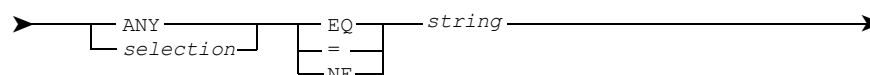
where:

m and p are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

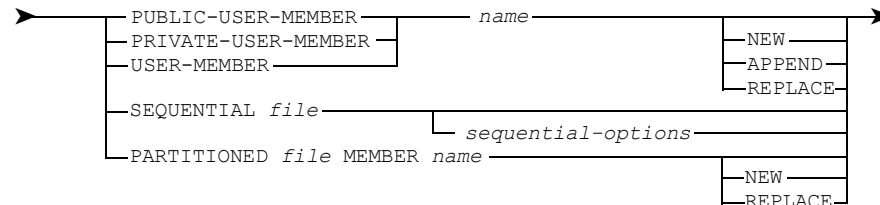
condition is:



where:

selection and *string* are as defined above.

destination is:

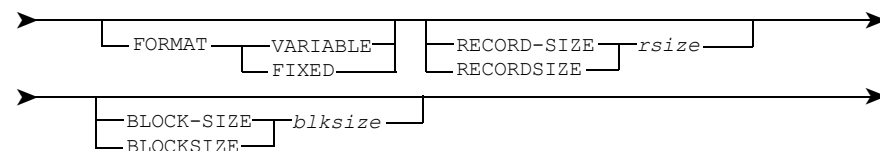


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



where:

rsize is the record length.

blksize is the block size.

DB2 LABEL

See ["DB2 COMMENT and DB2 LABEL" on page 40](#).

DB2 LIST CYCLES

To identify the cycles found in the DB2 design present in the Workbench Design Area (WBDA) and to list the tables which appear within each cycle.



DB2 LIST TABLES

To produce a list of all or some of the tables appearing in the DB2 design generated in the Workbench Design Area (WBDA).

```

>> DB2 LIST TABLES [ALPHABETICALLY]
[table-type] ; [.]

```

where *table-type* is:

```

> [ROOTS] [PARENTS] [LEAFS]
  [LEAVES]
> [DEPENDENTS] [INDEPENDENTS]

```

DB2 PLOT CLUSTER

To produce a DB2 Cluster Plot of all or some of the tables in the DB2 design.

```

>> DB2 PLOT CLUSTER [ALL]
  [NAMES name-list]
  [NUMBERS range-list]
[ALPHABETICALLY]
[USING FORMAT format-member] ; [.]

```

where:

name-list is a list of validly named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:

```

> m [TO n]

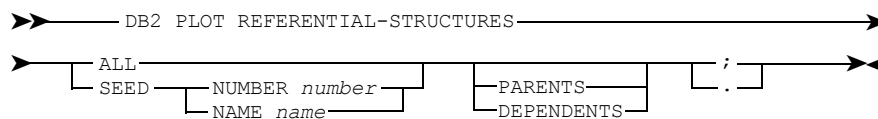
```

where *m* and *n* are valid WBDA table numbers and *n*, if it appears, is greater than *m*.

format-member is the name of a previously defined, valid format member.

DB2 PLOT REFERENTIAL-STRUCTURES

To produce a DB2 Referential Structures Plot of one or all of the referential structures in the DB2 design.



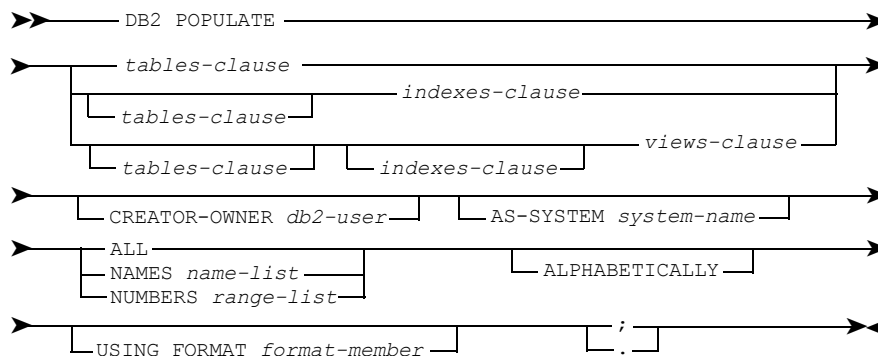
where:

number is a table number in the Workbench Design Area.

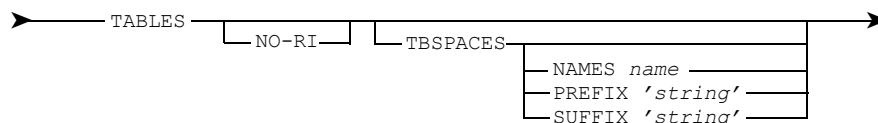
name is a table name.

DB2 POPULATE

To populate the repository with the results of the preceding PREVIEW IMPORT command.



where *tables-clause* is:

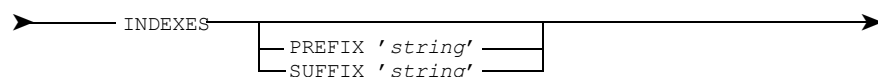


where:

name is an alphanumeric string of up to 32 characters which should conform to the rules for a valid Manager Products repository member name.

string is an alphanumeric string of up to 31 characters which should conform to the rules for a valid Manager Products repository member name.

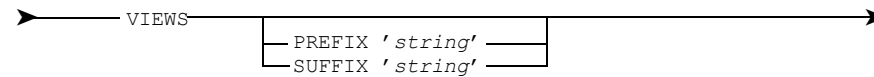
indexes-clause is:



where:

string is defined as above.

views-clause is:



where:

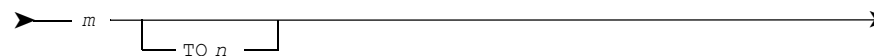
string is defined as above.

db2-user is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

system-name is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

name-list is a list of validly named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:

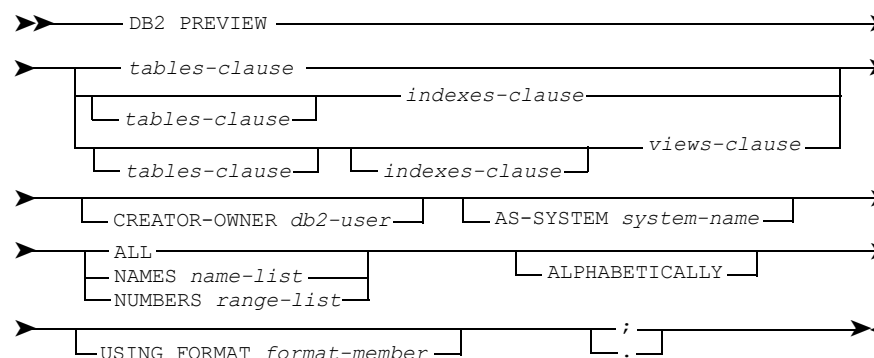


where *m* and *n* are valid WBDA table numbers and *n*, if it appears, is greater than *m*.

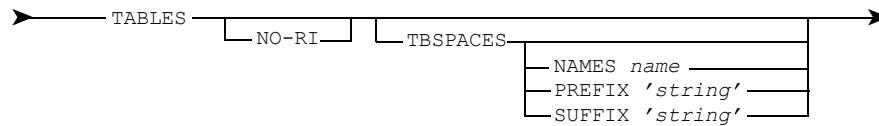
format-member is the name of a previously defined, valid format member.

DB2 PREVIEW

To generate and report repository member definitions from DB2 tables, indexes, and views in the DB2 design.



where *tables-clause* is:

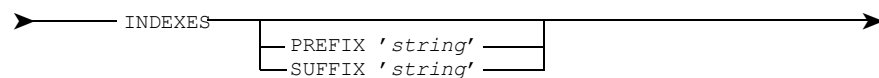


where:

name is an alphanumeric string of up to 32 characters which should conform to the rules for a valid Manager Products repository member name.

string is an alphanumeric string of up to 31 characters which should conform to the rules for a valid Manager Products repository member name.

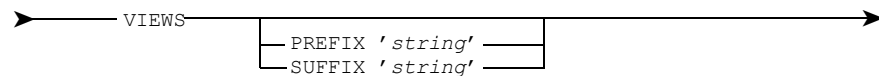
indexes-clause is:



where:

string is defined as above.

views-clause is:



where:

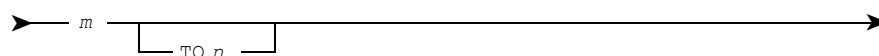
string is defined as above.

db2-user is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

system-name is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

name-list is a list of validly named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:

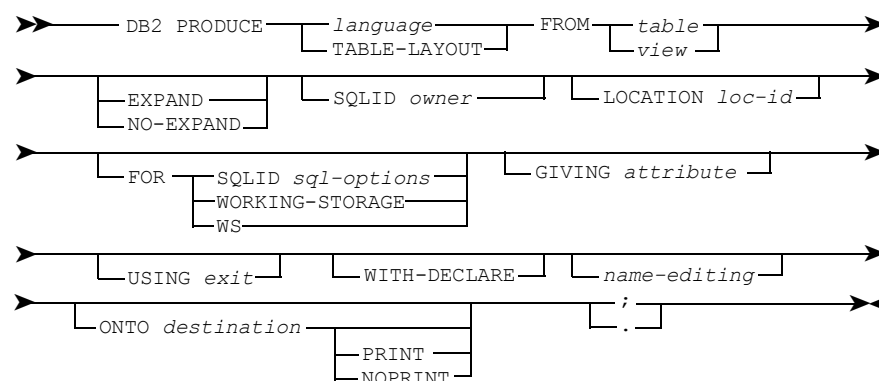


where m and n are valid WBDA table numbers and n , if it appears, is greater than m .

format-member is the name of a previously defined, valid format member.

DB2 PRODUCE

To generate either a host language data structure or a table layout from a DB2-TABLE or DB2-VIEW repository member.



where *language* is:



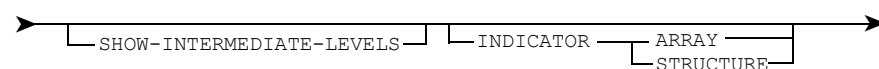
table is the name of a DB2-TABLE repository member.

view is the name of a DB2-VIEW repository member.

owner is a delimited string of up to 8 characters, giving the authorization ID of a specific user (for an SQL DECLARE statement).

loc-id is a delimited string of up to 16 characters, giving the name of a DB2 location (for an SQL DECLARE statement).

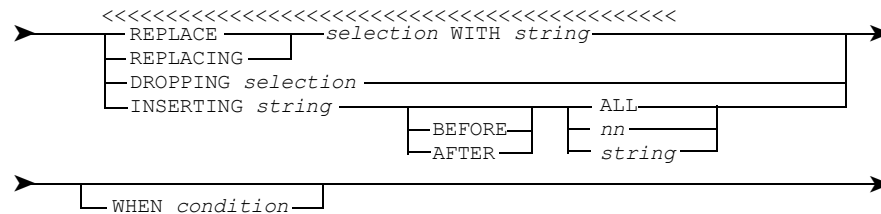
sql-options are:



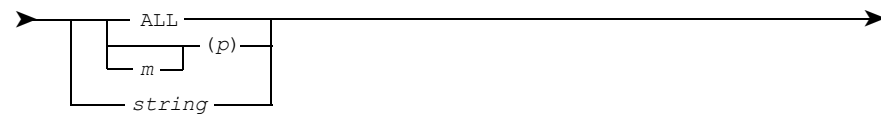
attribute is any text attribute.

exit is the name of an Executive Routine.

name-editing is:



where *selection* is:



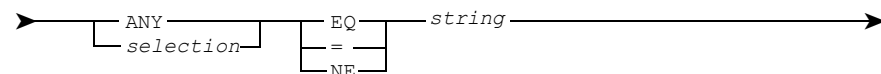
where:

m and p are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

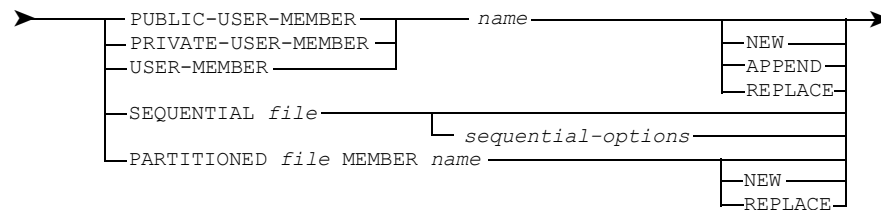
condition is:



where:

selection and *string* are as defined above.

destination is:

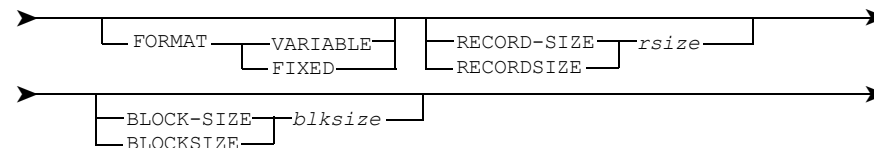


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



where:

*rsiz*e is the record length.

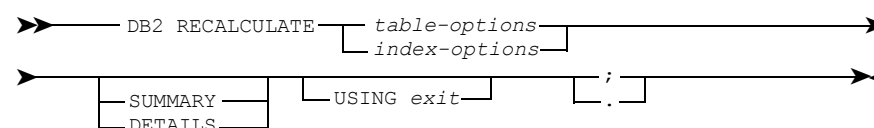
blksize is the block size.

DB2 REBIND

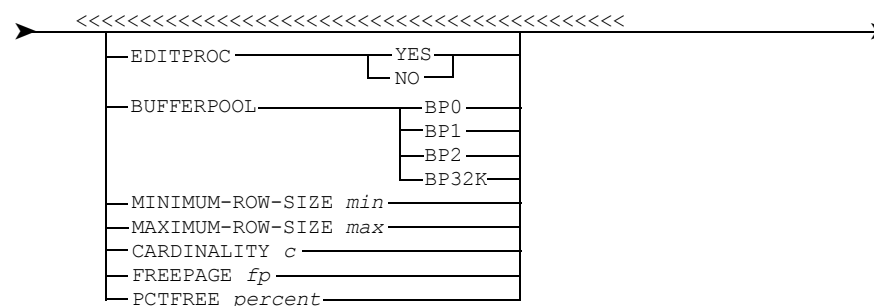
See ["DB2 BIND and REBIND" on page 32.](#)

DB2 RECALCULATE

To recalculate sizes of tables or indexes.



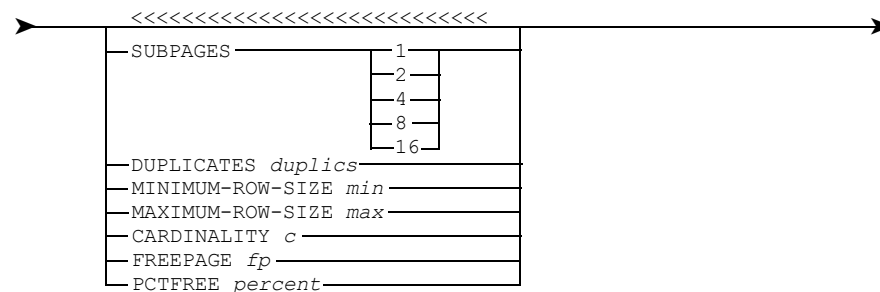
where *table-options* are:



where:

min, *max*, *c*, *fp*, and *percent* are integers, *percent* being between 0 and 100.

index-options are:



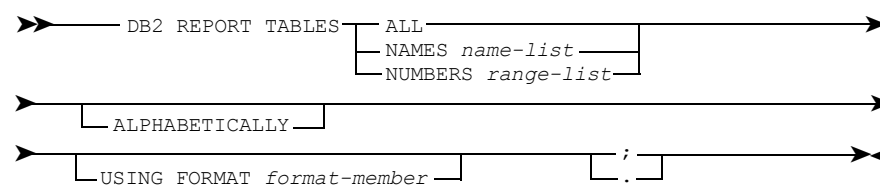
where *duplics*, *min*, *max*, *c*, *fp*, and *percent* are integers, with *percent* being between 0 and 100.

Note:

For *table-options* and *index-options* do not use the same keyword more than once.

DB2 REPORT

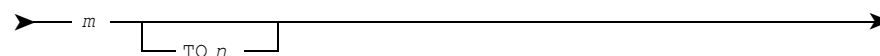
To produce a DB2 Table Report of all or some of the tables in the DB2 design.



where:

name-list is a list of validly named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:



where m and n are valid WBDA table numbers and n , if it appears, is greater than m .

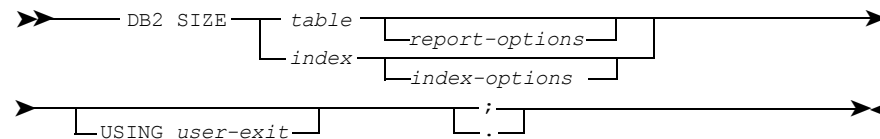
format-member is the name of a previously defined, valid format member.

DB2 REVOKE

See ["DB2 GRANT and DB2 REVOKE" on page 48](#).

DB2 SIZE

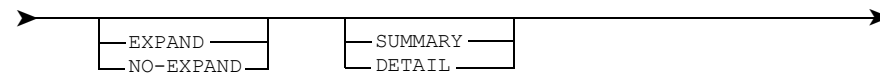
To calculate the size of a DB2 table or an index from its repository member definition.



where:

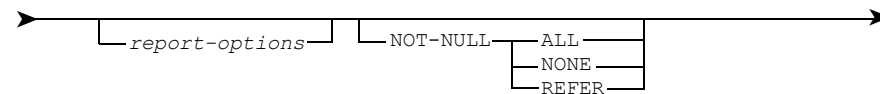
table is the name of a DB2-TABLE repository member.

report-options are:



index is the name of a DB2-INDEX repository member.

index-options are:



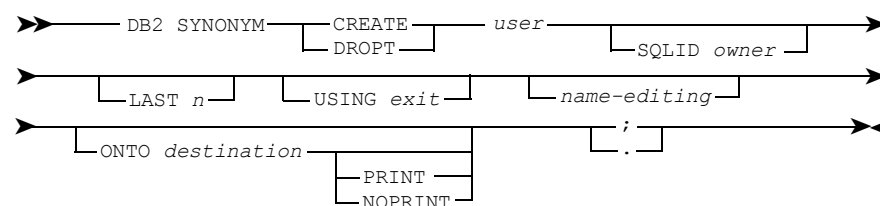
where:

report-options are as defined above.

user-exit is the name of an Executive Routine.

DB2 SYNONYM

To generate SQL CREATE SYNONYM or DROP SYNONYM statements.



where:

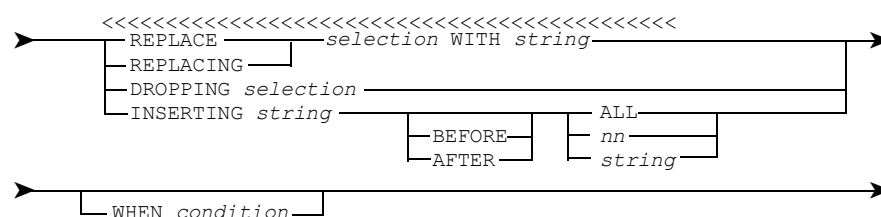
user is the name of a DB2-USER member.

owner is a delimited string of up to 8 characters, giving the authorization ID of a specific user.

n is an integer specifying how many synonyms are to be selected.

exit is the name of an Executive Routine.

name-editing is:



where *selection* is:



where:

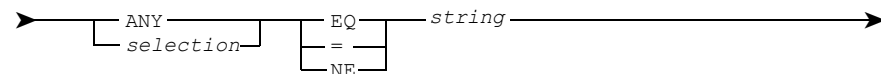
m and p are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

string is as defined above.

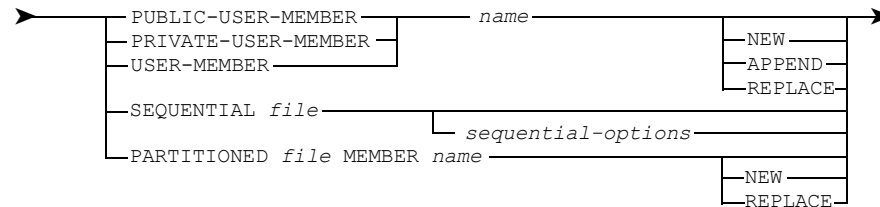
condition is:



where:

selection and *string* are as defined above.

destination is:

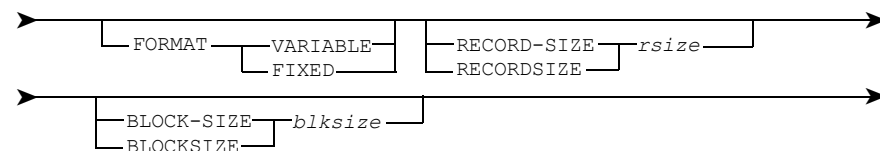


where:

name is the name of a USER-MEMBER.

file is the name of a sequential or partitioned dataset.

sequential-options are:



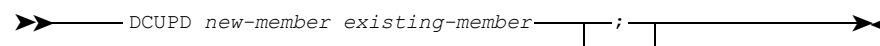
where:

rsiz is the record length.

blksize is the block size.

DCUPD

To create a new DOCUMENT member and copy into its CONTENTS attribute information from an existing member.



where:

new-member is the name of a new DOCUMENT member.

existing-member is the name of an existing DOCUMENT or LIFE-CYCLE-OBJECT-TYPE member.

DD

See ["D and DD Line Commands" on page 27](#).

DEACTIVATE

See ["DACT and DEACTIVATE" on page 28](#).

DELETE

To delete the Current Line of an Edit or Update buffer.

➤➤ — DELETE — ; ➤➤

where n is an integer from 1 to 9999.

DESIGN

To normalize the dependencies in the Workbench Design Area (WBDA) and generate the relations and records of the conceptual schema.

➤➤ — DESIGN — ➤➤

➤ — — AUDIT — ➤

➤ — ; ➤➤

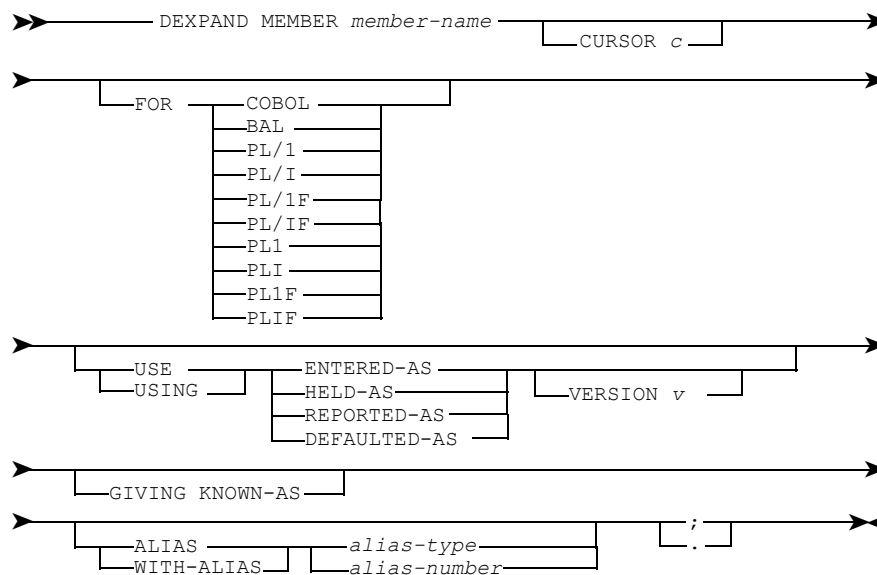
where *format-selection* is:

➤ — SUMMARY — — USING FORMAT *format-name* — ➤

where *format-name* is the name of a FORMAT member of the modeling repository.

DEXPAND Executive Command

To instigate the expansion of a named member, that is, to generate a data structure for the named member.



where:

c is an integer in the range 1 to 32767.

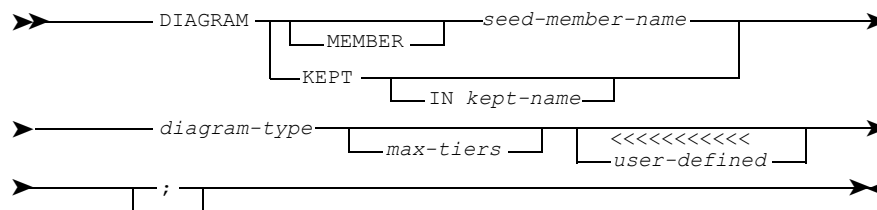
v is an unsigned integer specifying a number from 1 to 15.

alias-type is a keyword from the alias-type keyword list of the repository.

alias-number can be any number from 1 to the maximum number of general aliases allowed in your repository.

DIAGRAM

To generate a ASG-ManagerView (herein called ManagerView) diagram-member on your Corporate Repository



where:

seed-member-name is the name of an encoded member on your Corporate Repository, which is to be the seed, or starting point, for the diagram-member you wish to generate.

kept-name is the name of a kept-data-list containing the names of multiple seeds.

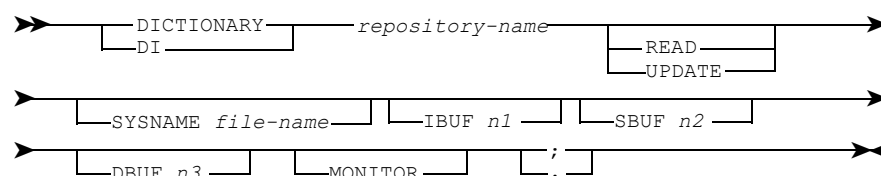
diagram-type is a mnemonic for a particular diagram-type.

max-tiers is an optional integer, which specifies how many levels (tiers) of relationships are to be followed from the seed or seeds in attempting to construct the diagram.

user-defined is an optional user-defined parameter which can be present up to six times.

DICTIONARY

To open a repository, to invoke the datasets and alter their buffer pool size, and to switch on buffer steal monitoring.



where:

repository-name is the name of a repository that has been established by the Controller.

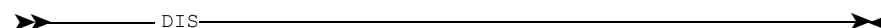
file-name is the logical file name (ddname) used in OS job control statements to indicate the external dataset name (physical file name) of the repository's index dataset.

n1, *n2*, and *n3* are unsigned integers in the range 2 to 32,000.

DIS and DISPLAY

To display a repository member's source record.

Line command:



Primary command:

➤ `DISPLAY member-name` `[;]` ➤

DL/I LIST

To obtain a list of DL/I root segments.

➤ `DL/I` `LIST ROOTS` `[SEGMENTS]` `[;]` ➤

DL/I
DLI
DL/1
DL1
IMS

DL/I MAP

To display:

- One or more diagrams of the DL/I segments generated
- The types of relationships that link them

➤ `DL/I` `MAP` `[PHYSICAL]` `[HIERARCHY]` `[UNIDIRECTIONAL-RELATIONSHIPS]` `[ACCESS-PATHS]` `[;]` ➤

DL/I
DLI
DL/1
DL1
IMS

DL/I REPORT

To produce:

- A report of the secondary key access paths derived from input userviews

or

- A report on all the segments generated

➤ `DL/I` `REPORT` `[SECONDARY-ACCESS-PATHS]` `[SEGMENTS]` `[;]` ➤

DL/I
DLI
DL/1
DL1
IMS

DMS

To provide a temporary repository management software mode for the duration of the current command.

➤ — DMS *command* — [] ; — ➤

where *command* is the name of a Manager Products command.

DO and END Directives

To group a sequence of instructions into a block. The block is executed zero or more times.

➤ — DO — [*array-name* ()] —

FOR <i>n</i>
UNTIL <i>condition</i>
WHILE <i>condition</i>

➤ — <<<<<<<<<<<< *instruction* — ➤

➤ — END — ➤

where:

array-name is the name of an array.

n is greater than or equal to zero.

condition is:

➤ — *clause* —

<<<<<<<<<<<<
AND — <i>clause</i>
OR —
XOR —
& —
—
&& —

➤ —

where *clause* is:

➤ — *expression* —

=
>=
>
<=
<
=
EQ
GE
GT
LE
LT
NE

➤ — *expression* — ➤

where *expression* is an expression subject to Full Evaluation.

DOC

To process a DOCUMENT member.

Primary command:

DOC *document-name* ;

Line command:

DOES

To ascertain whether a particular member of the repository uses (directly or indirectly) another specified member of the repository.

Diagram illustrating the use of member-name-1 and member-name-2. The top part shows a sequence of actions: "DOES member-name-1" followed by a branch. One branch leads to "USE member-name-2" which then branches into "DIRECTLY" and "INDIRECTLY". The other branch from "DOES member-name-1" leads to "DIRECTLY" and "INDIRECTLY" which then leads to "USE member-name-2". The bottom part shows a sequence of actions: a bracketed section followed by a semicolon ";" and then another bracketed section.

where *member-name-1* and *member-name-2* are the names of repository members. *member-name-1* must be encoded.

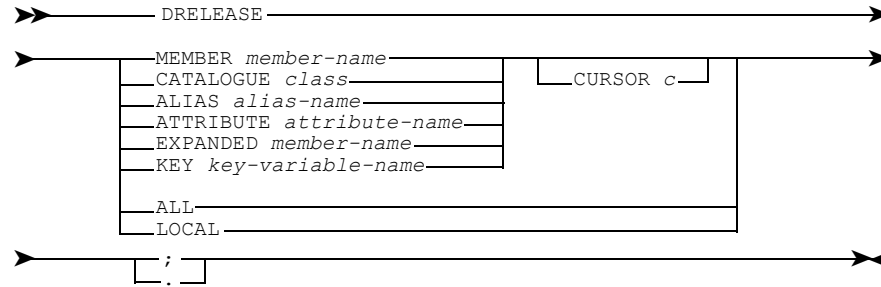
DOWN

To scroll forward one or more lines in a buffer (synonymous with NEXT).

where n is the number of lines to be scrolled forward in a buffer.

DRELEASE Executive Command

To nullify variables generated for a DACCESSED index-name or DEXPANDED member and release the index-name and all the variables associated with it from virtual storage.



where:

member-name is the name of a member on the Manager Products repository.

class is a catalog classification.

alias-name is the name of an alias.

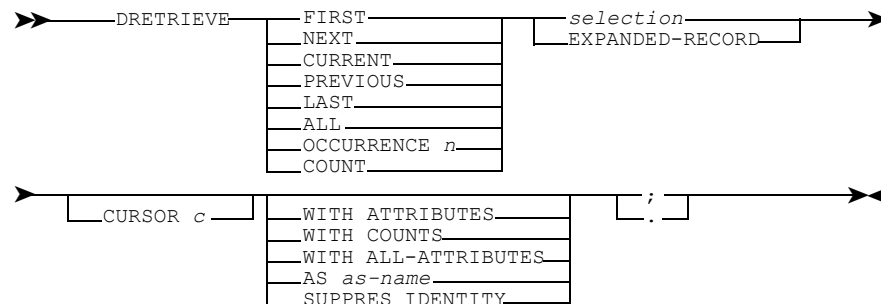
attribute-name is the name of an indexed attribute.

key-variable-name is the name of a numeric key identifying a member, alias, catalog, or indexed attribute on the Manager Products repository.

c is an integer in the range 1 to 32767.

DRETRIEVE Executive Command

To retrieve the occurrences of repeating clauses of a DACCESSed repository member.



EDDATE Function

To reformat the Manager Products standard date format into your installation's standard format, as defined in the DCUST installation macro.

➤➤ ——— EDDATE x (*expression*) ————— ➤➤

where x is I or O representing input and output format respectively.

EDIT

To create an MP-AID USER-MEMBER or edit an existing one.

➤➤ ——— EDIT ——— $mp-aid-member-name$; ————— ➤➤

where *mp-aid-member-name* is the name of a USER-MEMBER.

EDTIME Function

To reformat the Manager Products standard time format into your installation's standard format, as defined in the CUST installation macro.

➤➤ ——— EDTIME n (*expression*) ————— ➤➤

where n is I or O representing input and output format respectively.

ENCODE

To generate an encoded record from a member's source record and insert this encoded record into the repository's Data Entries dataset.

➤➤ ——— ENCODE ^{<<<< , <<<<}
member-name ————— ; ————— ➤➤

where *member-name* is the name of a repository member.

END Command

See ["LOGOFF and END" on page 94.](#)

END Directive

See ["DO and END Directives" on page 66.](#)

ENVIRONMENT

To give details of the Manager Products at your installation.

```

>>-----ENVIRONMENT----->
                ALL
                CUST
                ;

```

ERASE KEPT-DATA

To remove KEPT-DATA lists from virtual storage.

```

>>-----ERASE KEPT-DATA----->
>----->
    UNNAMED
    AND LIST list-name
    <<< , <<<
    LISTS list-name
    ALL
    ;

```

where *list-name* is the name of a KEPT-DATA list.

EXECUTE

To execute a User Executive Routine.

```

>>-----EXECUTE executive-routine-call----->
                ;

```

where *executive-routine-call* is the name of a User Executive Member on the MP-AID.

EXIT Directive

To cause an immediate exit from the current Executive Routine and Set return value(s) for the calling Executive Routine (the Executive Routine to which control returns).

```

>>-----EXIT----->
    expression
    EXEC
    SESSION

```

where *expression* is an expression subject to Full Evaluation that evaluates to an integer between 0 and 255 inclusive.

pds-options are:



where:

member is the name of a partitioned dataset member.

store-rules are as defined above.

store-var is the name of the variable into which extracted is stored.

mpr-command is a delimited string, giving any Executive Routine or Manager Products Primary Command (except EXTRACT).

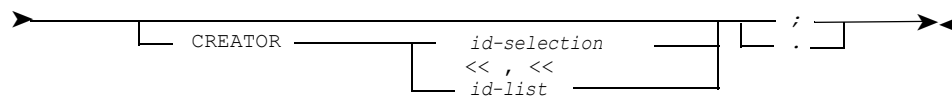
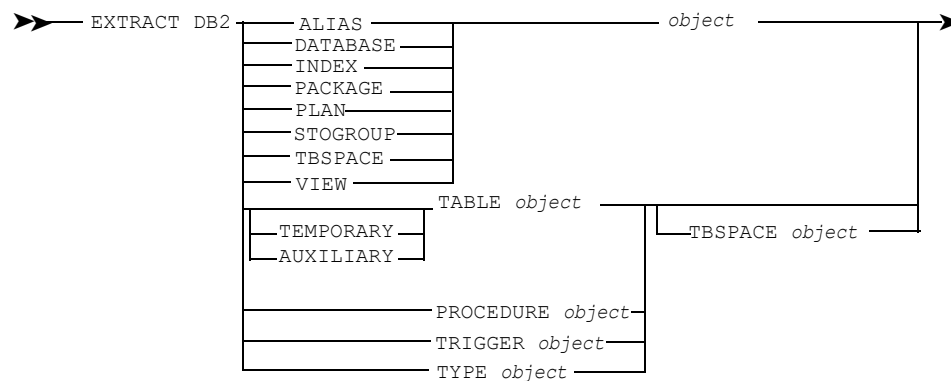
input-var is the name of the variable from which input is to be taken.

store-var, *store-rules* are as defined above.

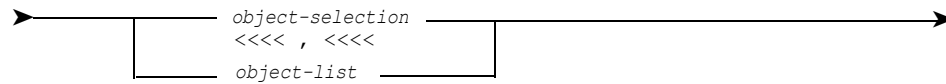
extract-routine is the name of an Executive Routine.

EXTRACT DB2

EXTRACT DB2 imports information about DB2 objects from your DB2 environment.



where *object* is:



where:

object-selection is a combination of characters and ? and * symbols, matching the name of a selection of external objects.

object-list is a list of one or more names of objects to be extracted.

where "?" represents any single character and "*" represents a string of (any number of) any characters.

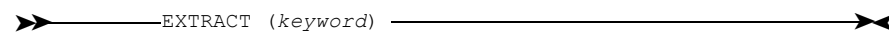
id-selection is a combination of characters and "?" and "*" symbols matching the authorization IDs of a selection of owners.

id-list is a list of one or more names of IDs to be extracted.

where "?" and "*" are as defined above.

EXTRACT Function

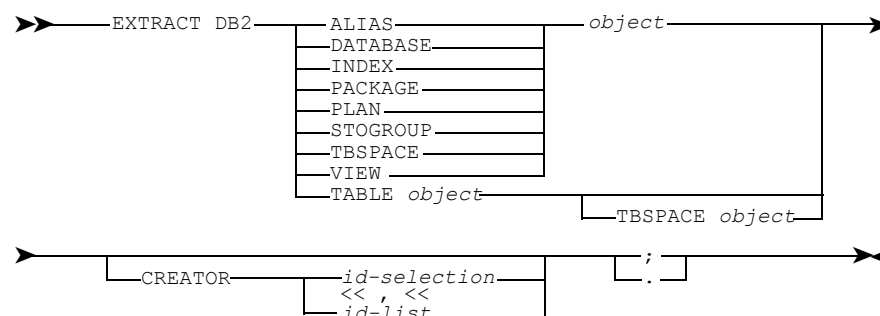
To return information about your Manager Products environment.



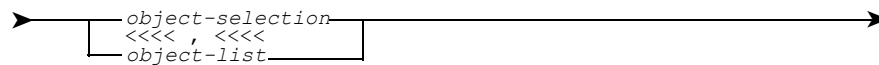
where *keyword* is one of CMD, CMRREL, CPUTIME, DFREE, DMRREL, DSN, DSRREL, DYRREL, ELEVEL, ENAME, ETYPE, EYRREL, IFREE, LFREE, MFREE, MMRREL, NOPRINT, PAGELN, PROFID, SFREE, SMRREL, SRBTIME, or VERB.

EXTRACT DB2

To import information about DB2 objects from your DB2 environment.



where *object* is:



where:

object-selection is a combination of characters and ? and * symbols, matching the name of a selection of external objects (where ? represents any single character and * represents a string of (any number of) any characters).

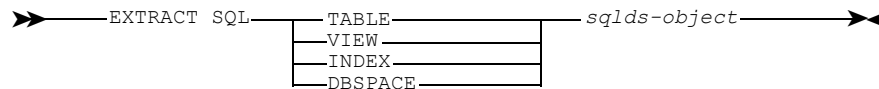
object-list is a list of one or more names of objects to be extracted.

id-selection is a combination of characters and ? and * symbols matching the authorization IDs of a selection of owners (where ? and * are as defined above).

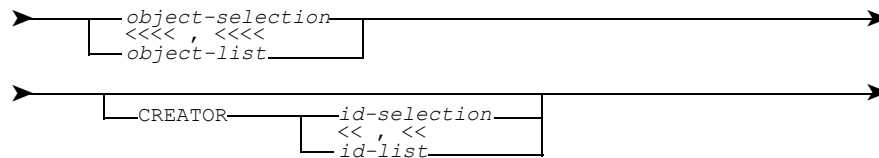
id-list is a list of one or more names of IDs to be extracted.

EXTRACT SQL

To import information from the SQLDS catalog into Procedures Language Global Variables in the Workbench Translation Area (WBTA).



where *sqlds-object* is:



where:

object-selection is a combination of characters and ? and * symbols matching the name of a selection of external objects (where ? represents any single character and * represents a string of (any number of) any characters).

object-list is a list of one or more names of objects to be extracted.

id-selection is a combination of characters and ? and * symbols matching the authorization IDs of a selection of owners (where ? and * are as defined above).

id-list is a list of one or more names of IDs to be extracted.

FDO Function

To return information about currently active blocks.

➤➤ — FDO (*keyword*) ————— ➤➤

where *keyword* is DARRAY, DFOR, or DLEVEL.

FDOC

To assemble and format a DOCUMENT member or to format the contents of an array, the output from a command, or the current buffer.

➤➤ — FDOC ————— ➤➤

<div style="display: flex; justify-content: space-between;"> MEMBER <i>member-name</i> </div>	;
<div style="display: flex; justify-content: space-between;"> ARRAY <i>array-name</i> CLASS <i>class-name</i> </div>	;
<div style="display: flex; justify-content: space-between;"> CMD <i>command</i> </div>	;

where:

member-name is the name of a DOCUMENT member.

array-name is the name of an array.

class-name is the class of variable to which the specified array belongs.

command is the command to be formatted. Multi-word commands must be enclosed in string delimiters.

FETCH KEPT-DATA

To retrieve KEPT-DATA lists stored on the MP-AID in KEPT-DATA members:

➤➤ — FETCH KEPT-DATA ————— ➤➤

<div style="display: flex; justify-content: space-between;"> AS <i>list-name</i> FROM <i>mpaid-name</i> </div>	;
--	---

➤➤ ————— ➤➤

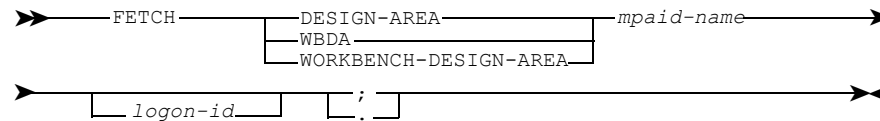
where:

list-name is the name of a KEPT-DATA list.

mpaid-name is the name of the KEPT-DATA member on the MP-AID in which the KEPT-DATA is stored.

FETCH WBDA

To retrieve the previously STORED content of the Workbench Design Area (WBDA) from a WORKBENCH member of the MP-AID and load it into virtual storage as the current WBDA content.



where:

mpaid-name is the name of a WORKBENCH member of the MP-AID.

logon-id is the Logon-id of the user who STOREd the MP-AID WORKBENCH member currently being FETCHed. If omitted, the default value is the logon ID of the current user.

FILE

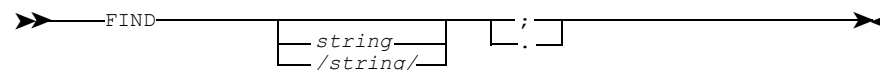
To enter the contents of an Update Buffer into the repository and generate an encoded record, or enter the contents of an Edit Buffer onto the MP-AID.



where *member-name* is the name of either a repository or an MP-AID member.

FIND

To find a string beginning in column one of a line, or in a column specified by an earlier ZONE command, and to scroll the text on the screen so that the line containing the string becomes the current line.



where:

string is a string of characters.

/string/ is a delimited string of characters.

FLUSH

To remove all data from the Workbench Design Area (WBDA).

```

>>-----FLUSH-----[DESIGN-AREA]-----[WBDA]-----[;]----->>

```

FORMAT

To:

- Generate output from the REPORT and BULK REPORT commands in a preferred alternative layout.
- Allow a user to generate the output from REPORT and BULK REPORT commands in user-defined layouts.

```

>>-----FORMAT-----[AS layout-name]----->>
                        [VIA]-----[MPAID]-----[mpaid-format-name]----->>
                        [MP-AID]-----[DICTIONARY format-name]----->>
>-----report-command----->
>-----[;]----->>

```

where:

layout-name is:

```

PSR-ITEMPSR-MODULEPSR-DETAIL
PSR-GROUPPSR-PROGRAMPSR-SUMMARY
PSR-FILEPSR-SYSTEM

```

mpaid-format-name is the name of a FORMAT member on the MP-AID.

format-name is the name of a FORMAT member in the repository.

report-command is a REPORT or BULK REPORT command.

FORWARD

To display the next screen in the current buffer.

```

>>-----FORWARD-----[;]----->>

```

FREE

To remove program code from virtual storage.

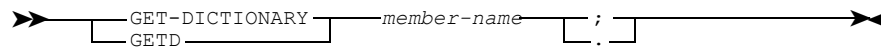
```

>>-----FREE-----[DESIGNMANAGER]-----[DSR]-----[;]----->>

```

GETD and GET-DICTIONARY

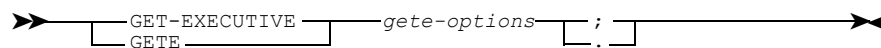
To copy a repository member into the current buffer.



where *member-name* is the name of a repository member.

GETE and GET-EXECUTIVE

To copy an EXECUTIVE member held on the primary MP-AID into the current buffer.



where *gete-options* are:



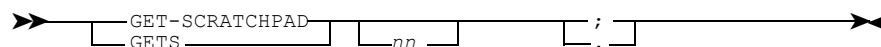
where:

executive is the name of an EXECUTIVE member on the MP-AID.

mpaid-name is the logical name of a secondary MP-AID.

GETS and GET-SCRATCHPAD

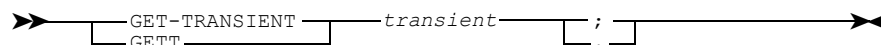
To insert the contents of the Scratchpad into the current Edit or Update Buffer.



where nn is the number of copies of the Scratchpad to be inserted.

GETT and GET-TRANSIENT

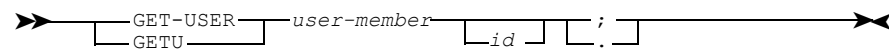
To copy a TRANSIENT member of the MP-AID into the current buffer.



where *transient* is the name of a TRANSIENT member on the MP-AID.

GETU and GET-USER

To copy a USER MEMBER into the current buffer.



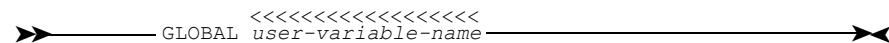
where:

user-member is the name of a USER-MEMBER on the MP-AID.

id is the Logon Identifier of a user owning a specified USER-MEMBER.

GLOBAL Directive

To declare Global Variables.

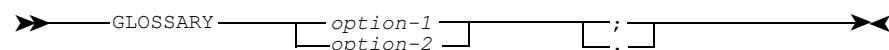


where *user-variable-name* is any valid user-defined variable name.

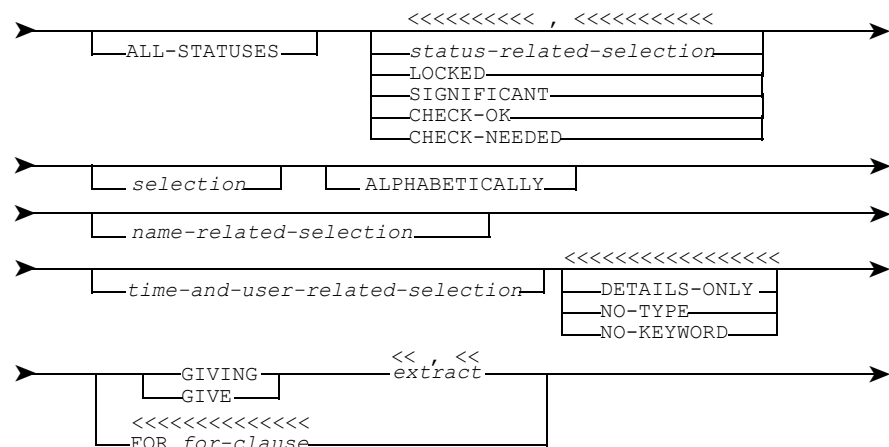
GLOSSARY

To display:

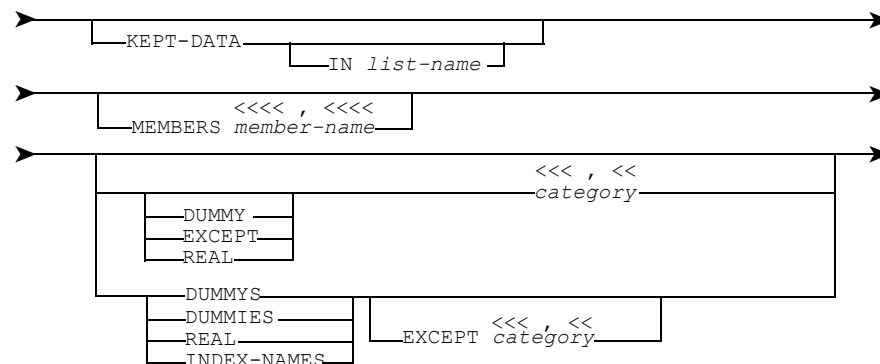
- Selected details of selected encoded members or dummies
- Selected aliases and selected details of encoded members that have those aliases



where *option-1* is:



where *selection* is:

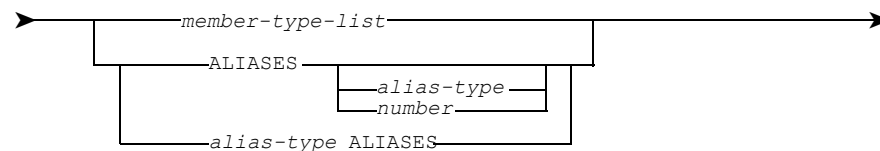


where:

list-name is the name of a KEPT-DATA list.

member-name is the name of a repository member.

category is:



where:

member-type-list is the interrogate keyword for any member type/collective member type in your repository.

Use the SHOW UDS command to list the member type interrogate keywords available in your repository.

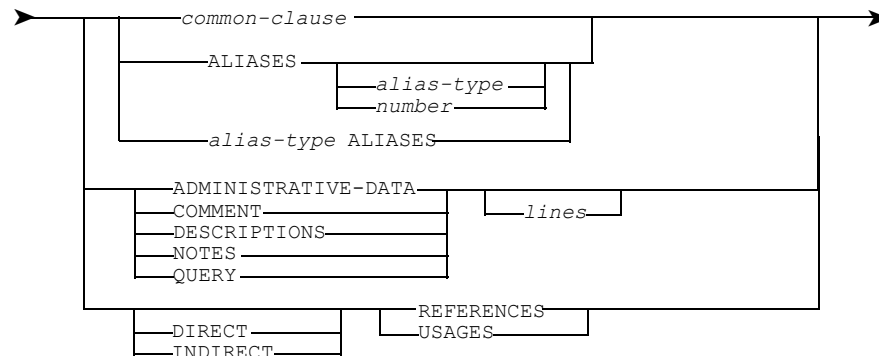
alias-type is one of the specific alias-type keywords available in your repository.

number is an integer from 1 to the maximum number of general aliases allowed in your repository.

Use the SHOW ALIAS-TYPES command to find out the alias-types and the number of general aliases available in your repository.

Status, *name*, *time*, and *user-related-selection* are defined in [Appendix A, "Name, Status, Time, and User-related Selection" on page 301](#).

extract is:

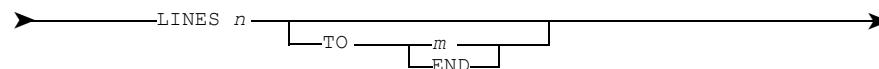


where:

common-clause is any clause available to all the member types present in your repository.

alias-type and *number* are as defined above.

lines is:



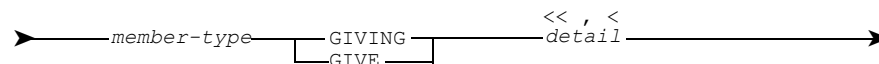
where:

m and *n* are unsigned integers in the range 1 to 32767, identifying delimited character strings.

To find out which common clauses are available in your repository, enter:

SHOW MEMBER-TYPE VARIABLES FOR COMMON-CLAUSES

for-clause is:



where:

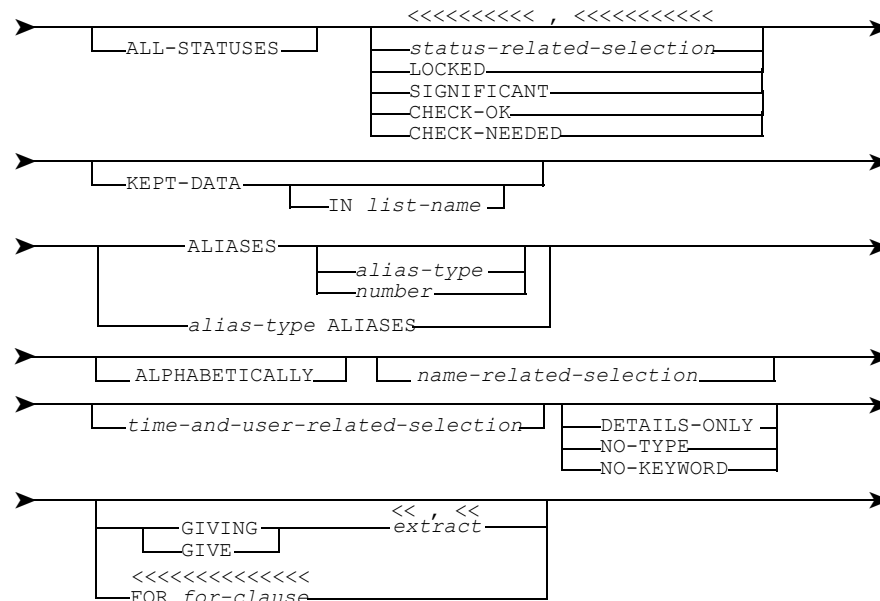
member-type is as defined above.

detail is *extract* as defined above, or any clause/attribute specific to the member type given in the FOR clause.

To find out which clauses or attributes are available to a particular member type in your repository, enter:

```
SHOW MEMBER-TYPE VARIABLES FOR MEMBER-TYPE member-type ;
```

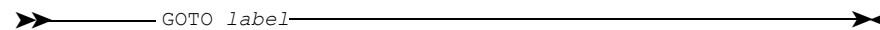
option-2 is:



where *status-related-selection*, *list-name*, *alias-type*, *number*, *name-related-selection*, *time-and-user-related-selection*, *extract*, and *for-clause* are as defined above.

GOTO Directive

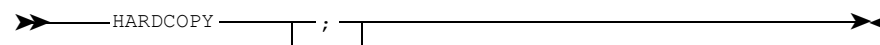
To cause control to be passed to the instruction following the specified label.



where *label* is a label name.

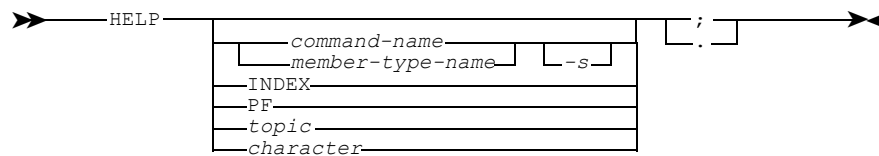
HARDCOPY

To print the contents of the current buffer on your printer.



HELP

To see the Infobank Help Entry Panel or request online help on a particular topic.



where:

command-name is the name of a Manager Products command.

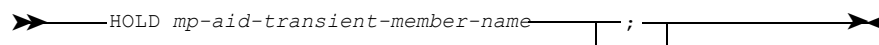
member-type-name is the name of a member type.

topic is a topic documented in InfoBank.

character is an alphabetic character.

HOLD

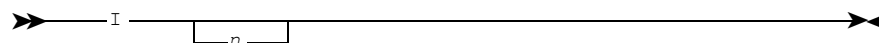
Store the contents of the currant buffer as a TRANSIENT member the MP-AID.



I Line Command

To:

- Insert n copies of the contents of the Scratchpad into a buffer after the line containing the I (Command Interface Environment).
- Insert an empty line into a buffer (Panel Interface Environment).



where n is an integer in the range 1 to 9999.

IEW EXPORT

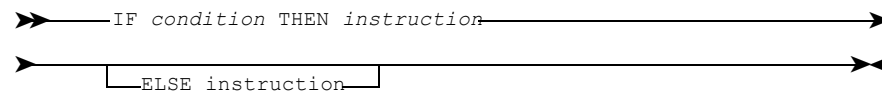
See ["ADW and IEW EXPORT" on page 7.](#)

IEW IMPORT

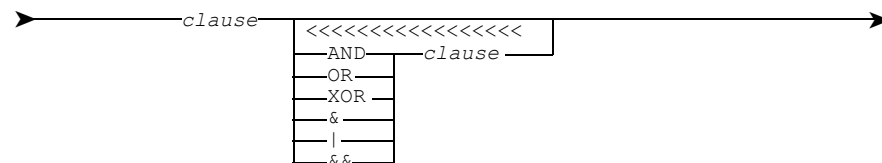
See ["ADW and IEW IMPORT" on page 7.](#)

IF Directive

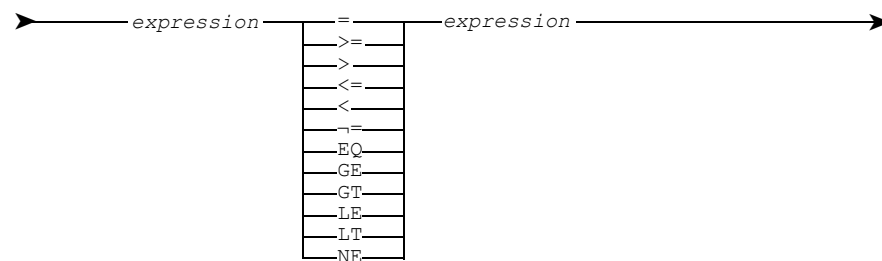
To conditionally execute instructions.



where *condition* is:



where *clause* is:



where:

expression is an expression subject to Full Evaluation.

instruction is a single command or directive, or a block of instructions.

IMS LIST

See ["DL/I LIST" on page 65](#).

IMS MAP

See ["DL/I MAP" on page 65](#).

IMS REPORT

See ["DL/I REPORT" on page 65](#).

INFOBANK

To display the Top Level Entry Panel to InfoBank.

```
➤➤ INFOBANK
  INFO-BANK ;
```

INSERT Function

To insert one string into another string.

```
➤➤ INSERT (new,target options)➤➤
```

where:

new and *target* are strings.

options is:

```
➤➤ , n , length , pad
```

where:

n and *length* are integers.

pad is a padding character.

INSERT Command

To insert a member's source record into the repository.

```
➤➤ INSERT member-name
  NUMBER integer  NOPRINT ;
                   NO-PRINT
                   PRINT
  member-definition
  ;
```

where:

member-name is the name of the repository member *integer* is an unsigned integer in the range of 1 to 10000, being the line number for the first line and the increment for subsequent line numbers of the member's source record.

member-definition is the member's source record.

INTERPRET Directive

To execute a dynamically constructed instruction.

INTERPRET

<<<<<<<<<

expression

where *expression* is an expression subject to Full Evaluation.

ISPF

To invoke ISPF ISPEXEC services.

```

  >> ISPF _____ ispf-service _____ ; >>
      GET  _____ ispf-name _____ procl-name _____
      PUT  _____ ispf-name _____ expression _____

```

where:

ispf-service is the name of an ISPF ISPEXEC service.

ispf-name is an ISPF variable name.

procl-name is a Procedures Language variable name.

expression is an expression subject to Limited Evaluation.

ISQL

To dynamically submit to your DB2 or SQL/DS environment SQL statements entered in the Command Area, printed in the current buffer, or filed in a USER-MEMBER.

```

graph LR
    ISQL[ISQL] --- Options["sql-statement  
user-member-name  
number  
DISCONNECT  
HELP  
sql/ds-sqlcode"]
    Options --- Semicolon[";"]
    Semicolon --- End[ ]

```

where:

sql-statement is any SQL statement that can be dynamically prepared for execution. The SQL statement can be a maximum of 255 characters long including leading, embedded, and trailing blanks. SQL SELECT statements must conform to the specifications of a full select statement.

user-member-name is the name of a USER-MEMBER in which an SQL statement is filed.

DISCONNECT disconnects the user from the current DB2 subsystem specified in MPDY42DFLT.

sql/ds-sqlcode is an SQL/DS SQLCODE number.

number is the number of rows in a result table to be printed in response to a SQL SELECT statement.

ITERATE Directive

To cause an immediate branch to the start of the currently active DO loop. If there is no currently active DO loop, ITERATE is invalid.

➤———ITERATE———➤

JUSTIFY

To justify text in an Update or Edit Buffer starting from the current line.

➤———JUSTIFY———*n1 n2* *n3* ; .———➤

where:

n1 is the column number for the left-hand margin within the range 1 to 236.

n2 is the column number for the right-hand margin within the range *n1*+19 to 255.

n3 is the amount of text you wish to justify. This may be:

- Text from the current line to the next completely blank line (enter no parameter)
- An exact number of lines starting from the current line (enter the number)
- Text from the current line to the end of the buffer (enter an asterisk)

K and KK Line Commands

To copy lines to the Scratchpad without deleting them from the current buffer and without overwriting the contents of the Scratchpad.

➤———K———*n*———➤

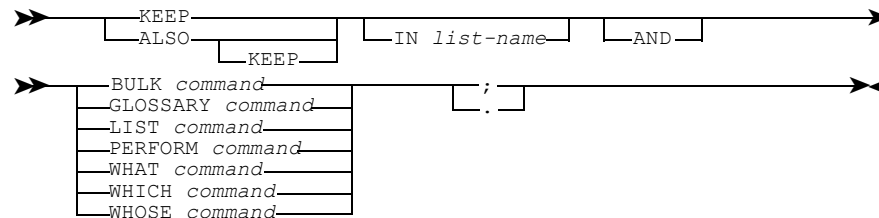
where *n* is an integer in the range 1 to 9999.

➤———KK———➤

➤———KK———➤

KEEP

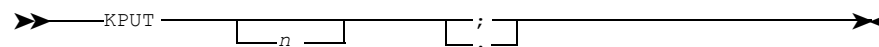
To create and accumulate, in a KEPT-DATA list, lists of member names, aliases, catalog classifications, and/or indexed attributes that can be used to drive further commands.



where *list-name* is the name of a KEPT-DATA list.

KPUT

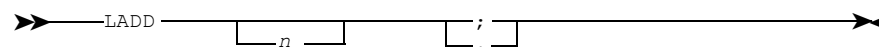
To copy the Current Line to the Scratchpad without deleting it from the current buffer, and without overwriting the contents of the Scratchpad.



where *n* is an integer in the range 1 to 9999.

LADD

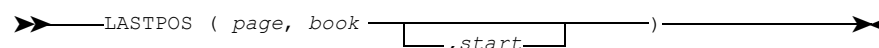
To add one or more blank lines to an Edit or Update Buffer immediately after the Current Line.



where *n* is an integer in the range 1 to 9999.

LASTPOS Function

To return the position of the last occurrence of one string in another string.



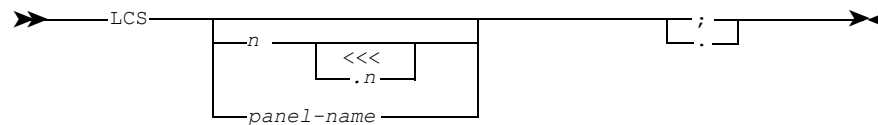
where:

page and *book* are strings.

start is an integer.

LCS

To start LifeCycle SERVICES.



where:

n is an option on a LifeCycle SERVICES menu or a ToolSet SERVICES menu.

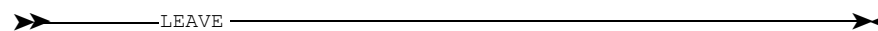
panel-name is the name of a LifeCycle SERVICES panel or a ToolSet SERVICES panel.

Note:

The LCS command can only be entered in the command interface environment.

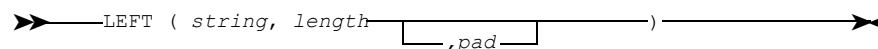
LEAVE Directive

To cause an immediate branch to the first instruction after the currently active DO loop. If there is no currently active DO loop, LEAVE is invalid.



LEFT Function

To return a string from the beginning of the given string.



where:

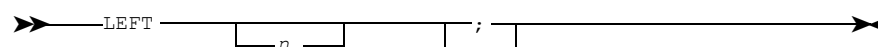
string is a string.

length is an integer.

pad is a character.

LEFT Command

To display text to the left of the text currently displayed.



where n is an integer in the range 1 to 254.

LENGTH Function

To return the length of a string.

➤ ———— `LENGTH(string)` ———— ➤

where *string* is a string.

LIST Data Modeling and Design Command

To produce listings of one or more categories of data present in the Workbench Design Area (WBDA).

➤ ———— `LIST` ———— `ALPHABETICALLY` `RECENT` `<<<, <<<` *selection* ———— ➤
 ➤ ———— `;` ———— ➤

where *selection* is:

➤ ————

USERVIEWS
ENTITLES
DATA-VIEWS
DATA-ELEMENTS
DEPENDENCIES
FDS
IMPLIED
MVDS
RELATIONS
RECORDS
ROOTS
RECORDS
DOMAINS

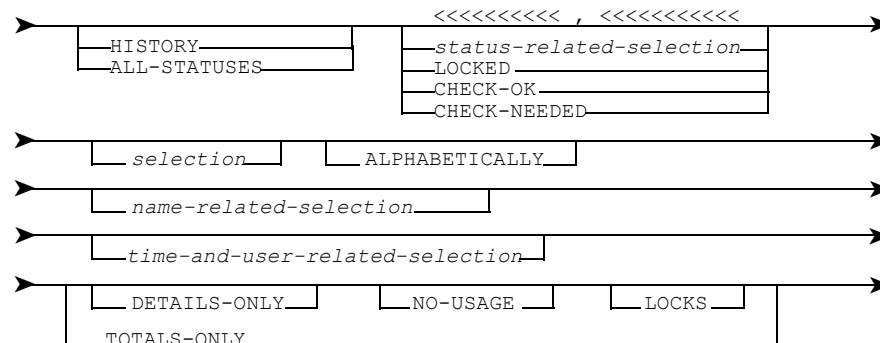
 ———— ➤

LIST

To list all or a selection of index-names.

➤ ———— `LIST` ———— `list-clause` ———— `LOCK-INFORMATION` ———— `<<<, <<` `MEMBERS` *mem-name* ———— `;` ———— ➤

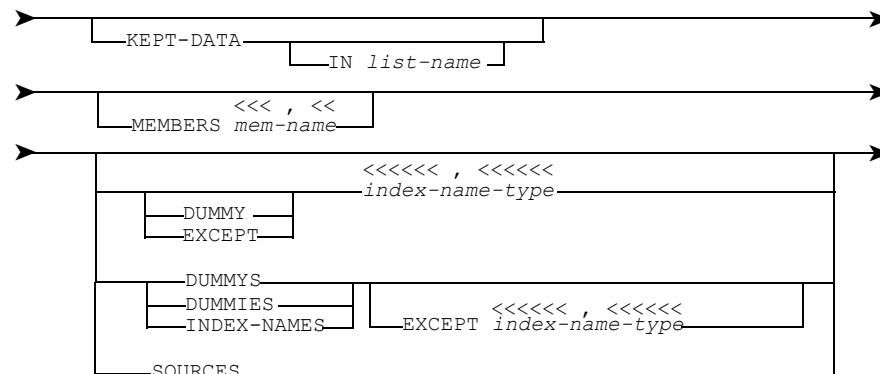
where *list-clause* is:



where:

status-related-selection is defined in [Appendix A, "Name, Status, Time, and User-related Selection" on page 301](#).

selection is:

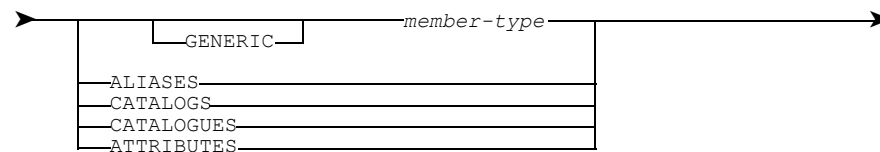


where:

list-name is the name of a KEPT-DATA list.

mem-name is the name of a repository member.

index-name-type is:



LOCK

To prevent concurrent updating of a repository member by other users.

```

>> LOCK NEW member-name ;
      EXISTING
      RENEW
      RELEASE
  
```

where *member-name* is the name of a repository member.

LOGOFF and END

To end a Manager Products run.

```

>> LOGOFF
    END ;
  
```

LOGON

To log on to Manager Products.

```

>> LOGON logon-identifier PASSWORD password
    RECONNECT ;
  
```

where:

logon-identifier and *password* are subject to rules governing names.

RECONNECT enables a user with an Exclusive Logon Profile to log on after the previous Manager Products session terminated abnormally, for example with an abend.

LOOKASIDE

See ["BROWSE and LOOKASIDE" on page 14](#).

LOWER Function

To return a string with all characters translated to lowercase.

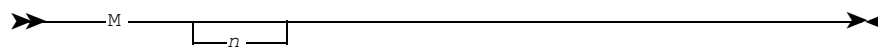
```

>> LOWER( string )
  
```

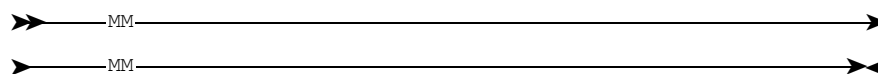
where *string* is a string.

M and MM Line Commands

To move lines to the Scratchpad from the current buffer, overwriting the contents of the Scratchpad.



where n is an integer in the range 1 to 9999.



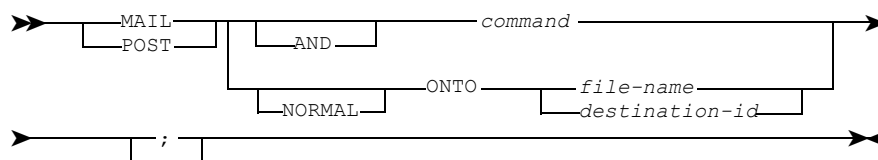
MAIL and POST

To:

- Write Manager Products command output to a sequential dataset.

or

- To specify the file that is to receive the output from subsequent MAIL or POST commands.



where:

command is the name of a command.

file-name is the logical name of the dataset that is to receive the output from subsequent POST/MAIL commands (that is, the dataset that is to be used as the current POST/MAIL Sequential Output Dataset). *file-name* can have a maximum of seven characters under DOS or a maximum of eight characters under OS.

file-name must not be any of these:

- MPRACWF
- MPRDIAG
- MPAID, MPAIDR, MPAIDV, or the name of any concatenated MP-AID
- The name of the repository, or the repository name with a suffix of: B, C, D, E, F, G, H, I, J, K, L, M, N, R, S, or V

destination-id is the logical name of the Extrapartition dataset that is to receive the output from subsequent POST/MAIL commands when operating under CICS. The name must have been specified in the CICS Destination Control Table as a four-character Destination Identification; *destination-id* is truncated from the right to four characters if it is declared as more than four characters in the POST/MAIL command.

MATRIX

To output a matrix report showing the relationship between members in two KEPT-DATA lists.

[illegible]

where:

list-name-1 and *list-name-2* are the names of KEPT-DATA lists.

margin is an integer specifying the number of character spaces in the margin in which the names of the members in *list-name-1* are listed.

column is an integer specifying the number of character spaces separating columns in the matrix.

attribute is any attribute that can establish a relationship between the members in *list-name-1* and *list-name-2*.

attribute-id is a single character which represents the attribute within the matrix.

MAX Function

To return the largest value from the list of passed arguments.

MEMTYPE Function

To return a form of a member's member-type.

MEMTYPEc(*member-type*)

where c is B, E, I, L, P, R, S, or X.

MER and MERGE

To load a repository member into your Workbench Design Area.

Panel Interface Environment Line Command:

```
➤➤——MER———→❏
```

Panel Interface Environment Primary Command:

```
➤➤——MERGE member-name———→❏
```

Command Interface Environment Primary Command:

```
➤➤——MERGE———→❏
      [UNCONDITIONALLY] [NO-VERIFY]
      <<<< , <<<<
➤——[EXPAND]——MEMBERS member-name———→❏
      [KEPT-DATA]——[IN kept-data-list-name]
➤——[;]———→❏
```

where:

member-name is the name of a verified USERVIEW, ENTITY, or VIEWSET member of the Modeling Repository.

kept-data-list-name is the name of a named KEPT-DATA list.

MERE

To load a repository member into your Workbench Design Area. The data groups belonging to the member are then expanded.

Primary Command:

```
➤➤——MERE member-name———→❏
```

Line Command:

```
➤➤——MERE———→❏
```

MERGE

To combine the source records of two or more ITEM members of the repository.

```
➤➤——MERGE target-name WITH <<<< , <<<< member-name———→❏
      [;]———→❏
```

where:

target-name is the name of an ITEM whose member definition is to be the definitive item for the data item.

member-name is the name of an ITEM whose member definition is to be subsidiary to the member definition held in *target-name*.

MERU

To load an unverified repository member into your Workbench Design Area, providing that the unverified member is not a dummy.

Primary Command:

➤—MERU *member-name* ————— □; □—————➤

Line Command:

➤—MERU —————➤

MERUE

To:

- Load an unverified repository member into your Workbench Design Area, providing that the unverified member is not a dummy.
- Expand the data groups belonging to the member.

Primary Command:

➤—MERUE *member-name* ————— □; □—————➤

Line Command:

➤—MERUE —————➤

MESSAGE

To output a Manager Products message.

➤—MESSAGE *message-number message-level variable-text* —————➤

where:

message-number is any integer below 65536.

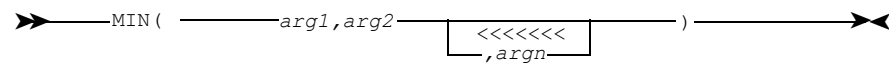
message-level is one of the following:

I or i	representing Informatory message
W or w	representing Warning message
E or e	representing Error message
S or s	representing Serious error message
C or c	representing Critical error message

variable-text can be specified 0 to 8 times, each text string containing up to 256 characters.

MIN

To return the smallest value from the list of passed arguments.

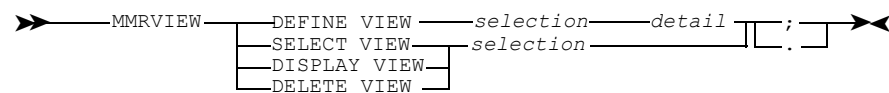


MM

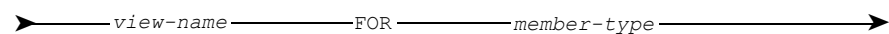
See ["M and MM Line Commands" on page 95](#).

MMRVIEW

To define, select, display, or delete a view on a member-type.



where *selection* is:

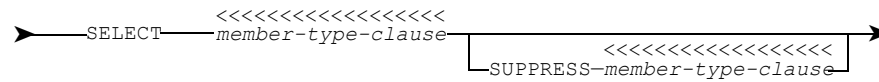


where:

view-name is the name of the view.

member-type is any valid number-type in your repository.

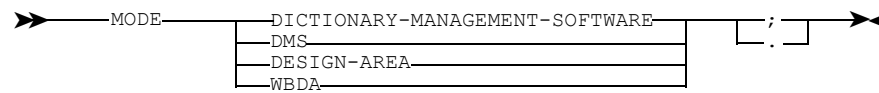
detail is:



where *member-type-clause* is any valid clause in the member-type.

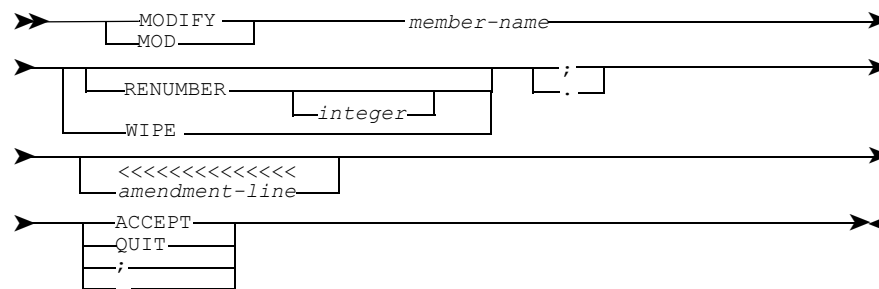
MODE

To set the environment to which a command should be routed, where ambiguity exists.



MODIFY

To modify the source record of an individual member and encode the modified source record.

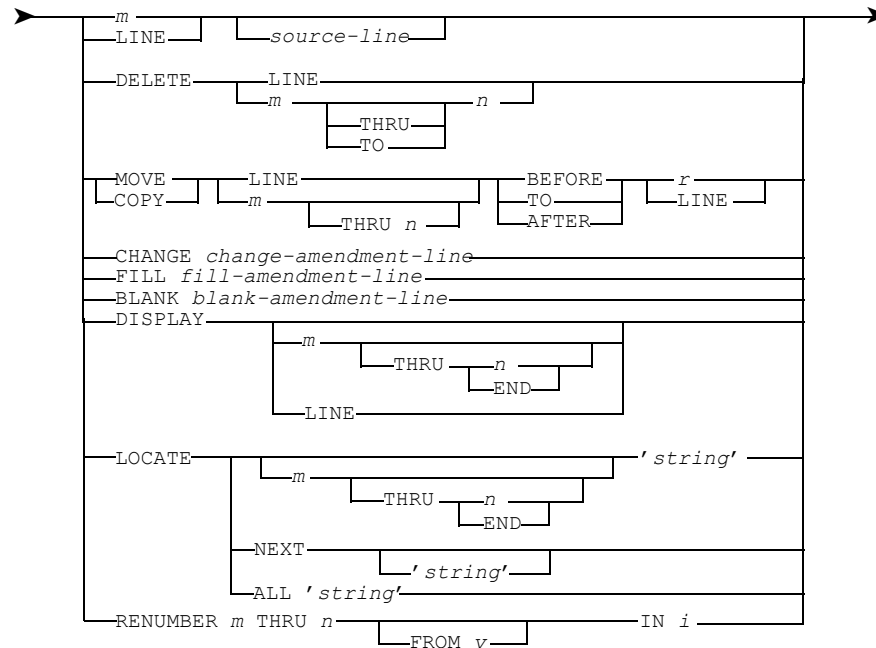


where:

member-name is the name of a repository member.

integer is an unsigned integer in the range 1 to 10000.

amendment-line is:



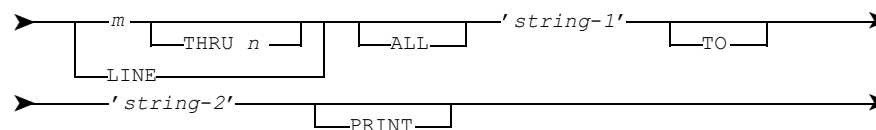
where:

m, *n*, *r*, and *v* are unsigned integers in the range 1 to 16777215, being line numbers.

i is an integer of increment.

source-line is a member definition statement line appropriate to the member-type.

change-amendment-line is:

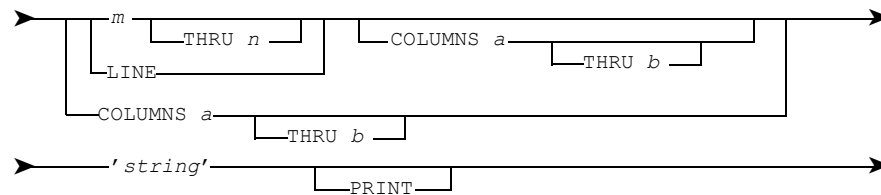


where:

m and *n* are as defined above.

string, *string-1*, and *string-2* are character strings from 1 to 254 characters.

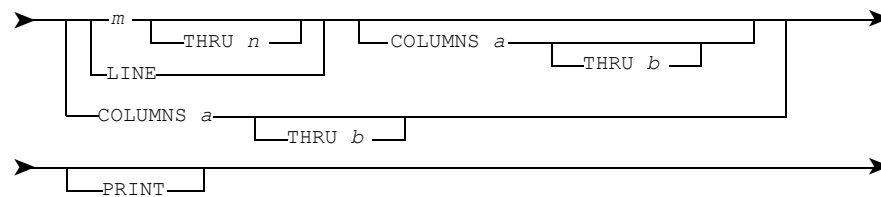
fill-amendment-line is:



where:

string, *m*, *n*, *a*, and *b* are as defined above.

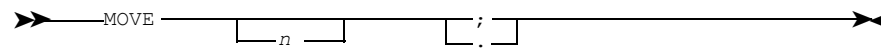
blank-amendment-line is:



where *m*, *n*, *a*, and *b* are as defined above.

MOVE

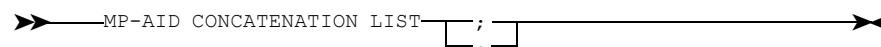
To remove lines, from the current line onward, of an Edit or Update Buffer and put them into the Scratchpad, overwriting the current contents of the Scratchpad in the process.



where *n* is an integer.

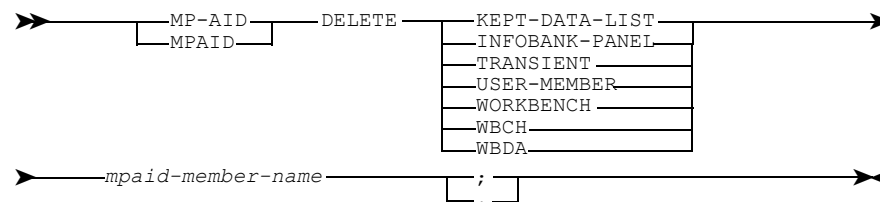
MP-AID CONCATENATION LIST

To list all your MP-AIDs.



MP-AID DELETE

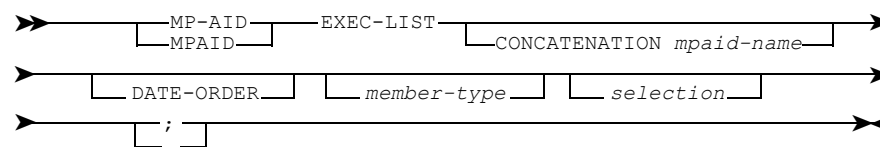
To delete a member on the MP-AID owned by your Logon Identifier.



where *mpaid-member-name* is the name of a member on the MP-AID.

MP-AID EXEC-LIST

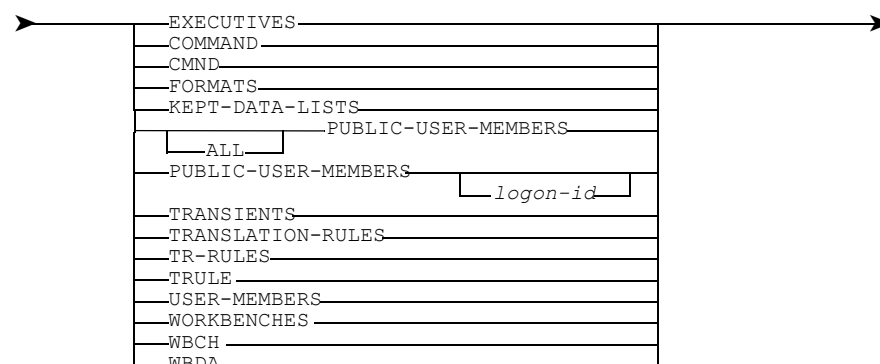
To list specified categories of MP-AID members for subsequent processing by an Executive Routine.



where:

mpaid-name is the logical name of a secondary MP-AID.

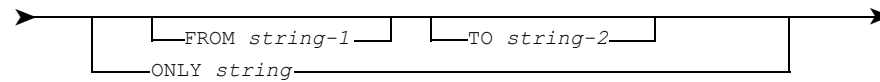
member-type is:



where:

logon-id is the logon identifier.

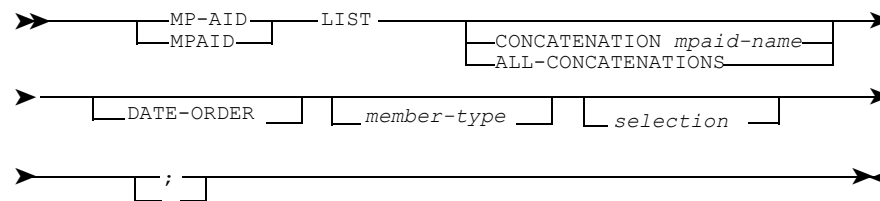
selection is:



where *string-1*, *string-2*, and *string* represent the range of the MP-AID members to be selected, and can be from 1 to 10 characters long.

MP-AID LIST

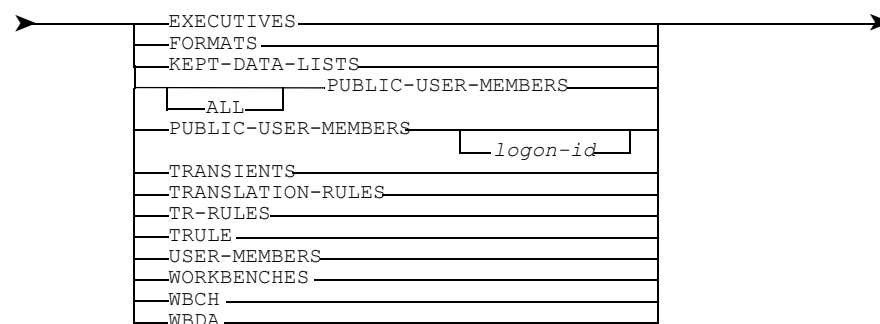
To list MP-AID members.



where:

mpaid-name is the logical name of a secondary MP-AID.

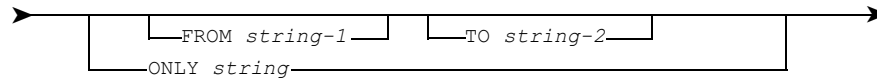
member-type is:



where:

logon-id is the logon identifier of a specific user.

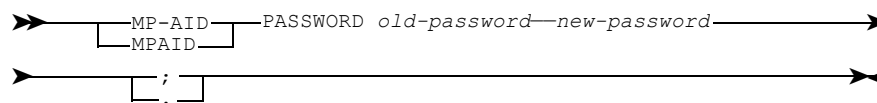
selection is:



where *string-1*, *string-2*, and *string* represent the range of the MP-AID members to be selected and can be from one to ten characters.

MP-AID PASSWORD

To change the password that you use in conjunction with your logon ID when logging on to Manager Products.



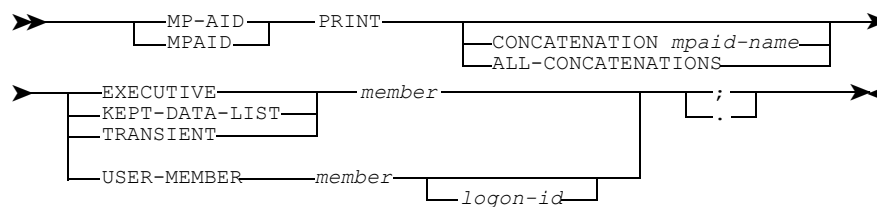
where:

old-password is the password used to log on to the current session.

new-password is the password to be used when logging on subsequently; the password may be a maximum of eight characters in length.

MP-AID PRINT

To print an MP-AID member.



where:

mpaid-name is the logical name of a secondary MP-AID.

member is the name of a member on the MP-AID.

logon-id is the name of a user's Logon Identifier.

MP-AID RESET USER-MEMBER

To reset the update-in-progress flag for an MP-AID USER-MEMBER created under the ownership of an exclusive Logon Profile, after an abnormal termination of a Manager Products session, so that the USER-MEMBER can be used again.

➤—MP-AID RESET USER-MEMBER *mpaid-member-name* ; ➤

MPMVGHTOC

To convert a USER member to a COMMAND member.

➤—MPMVGHTOC *hds-name* —————➤

where *hds-name* is the name of the USER member.

MPMVGHTOU

To convert an HDS-Table to a USER member.

➤—MPMVGHTOU *hds-name* —————➤

where *hds-name* is the name of the HDS-Table.

MPR Directive

To prefix Manager Products commands within Executive Routines.

➤—MPR— *Manager-Products-command* ; ➤

MPRAID Function

To return a numeric value which indicates whether the current top level Executive Routine was initiated by input from the Command Area and/or a PF key.

➤—MPRAID () —————➤

MPRCMPW Function

To return the password for the current Manager Products Logon session.

➤—MPRCMPW () —————➤

MPRDDPW Function

To return the password for the currently open repository, if any.

➤—MPRDDPW () —➤

MPRE Directive

To identify a Manager Products command using Full Evaluation instead of Limited Evaluation.

➤—MPRES *command* ; . —➤

where *command* is any Manager Products command.

MPRSU Function

To return one if the specified Selectable Unit or Functional Unit is present at your installation or zero if the Selectable Unit or Functional Unit is absent at your installation.

➤—MPRSU (*su-code*) —➤

where *su-code* is a four-character Selectable Unit or Functional Unit code.

MPRUCLS Function

To identify whether the user has the classification represented by the specified current-user-class.

➤—MPRUCLS (*current-user-class*) —➤

where *current-user-class* is A or a, C or c, G or g, M or m, U or u.

MPRUDSN Function

To return the UDST NAME of the currently open repository.

➤—MPRUDSN () —➤

MPXX Directive

To identify an Executive Routine and optionally define a unique literal delimiter and override renamed PRIMARY COMMANDs in corporate Executive Routines.

```
➤➤—MPXX—┐┐LITERAL=character┐┐DEFCMD┐┐➤➤
```

where *character* is any valid literal delimiter.

MTHelp

To display help on any member type available in your repository.

```
➤➤—MTHelp member-type —┐┐;┐┐➤➤
```

where *member-type* is any member type available in your repository.

MVW

To initialize a Manager Products environment for Workstation Mode or to re-establish a Manager Products environment on leaving Workstation Mode.

```
➤➤—MVW —┐┐ON┐┐┐┐OFF┐┐┐┐;┐┐➤➤
```

NAME

To assign names to relations or records in the Workbench Design Area (WDBA).

```
➤➤—NAME —┐┐RELATIONS┐┐name-clause┐┐;┐┐➤➤
┐┐RECORDS┐┐
```

where *name-clause* is:

```
➤➤┐┐KEY data-element-name┐┐AS name┐┐➤➤
┐┐NUMBER number┐┐
```

where:

data-element-name is the name of a data element in the Workbench Design Area which comprises all or part of the key of the relation or record being named.

number is the Workbench Design Area number of the relation or record currently being named.

name is assigned as a relation or record name depending on whether RELATIONS or RECORDS has been specified. It must conform to the rules for coding repository member names.

NDATE Function

To reformat the date from your installation's standard format, as defined in the DCUST installation macro, to the Manager Products standard date format.

➤➤—NDATE x (*expression*)—————➤➤

where x is I or O representing input and output format respectively.

NEXT

To move the window on the text displayed down n lines (synonymous with DOWN).

➤➤—NEXT— n —;.—————➤➤

where n is the number of lines to be scrolled forward in a buffer.

NOP Directive

To provide a dummy directive that has no effect.

➤➤—NOP—————➤➤

NOPRINT

To suppress the output of any Manager Products command. The command will be executed as normal.

➤➤—NOPRINT *command*—;.—————➤➤

where *command* is a Manager Products command.

NTIME Function

To reformat the date from your installation's standard format, as defined in the DCUST installation macro, to the Manager Products standard date format.

➤➤—NTIME n (*expression*)—————➤➤

where n is I or O representing input and output format respectively.

OVERLAY Function

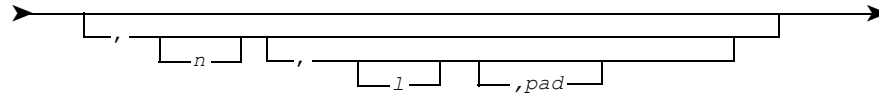
To return one string overlaid by another string.

➤➤—OVERLAY (*new,target,options*)—————➤➤

where:

new and *target* are strings.

options is:



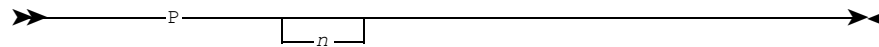
where:

n and l are integers.

pad is a character.

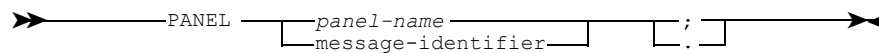
P Line Command

To insert n copies of the Scratchpad into an Edit or Update buffer preceding the line on which the P occurs.



PANEL

To display the InfoBank panel indicated by the panel-name or message-identifier.



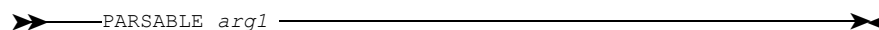
where:

panel-name is the name of an InfoBank panel.

message-identifier is the alphanumeric string which precedes any Manager Products message.

PARSABLE Function

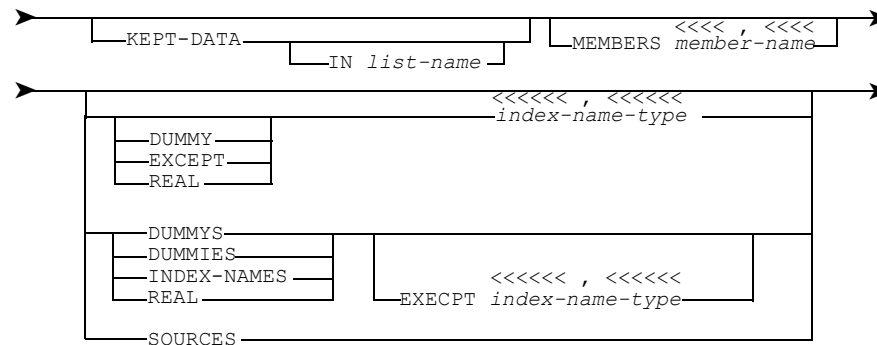
To return one if the string which results from the evaluation of *arg1* is parsable. If the string which results from *arg1* cannot be parsed, a value of zero is returned.



where *arg1* is a parameter.

status-related-selection is defined in [Appendix A, "Name, Status, Time, and User-related Selection" on page 301](#).

selection is:

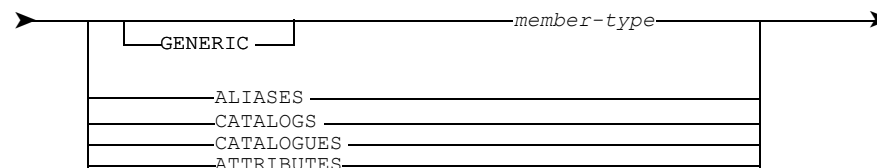


where:

list-name is the name of KEPT-DATA list.

member-name is the name of a repository member.

index-name-type is:

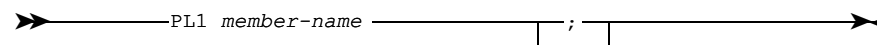


where *member-type* is the interrogate keyword for any member type, or any collective member type, available in your repository.

PL1

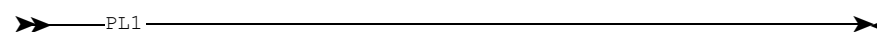
To generate a PL/I data structure.

Primary Command:



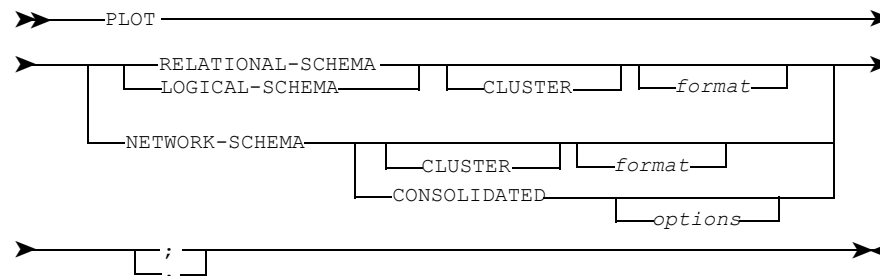
where *member-name* is the name of a repository member.

Line Command:

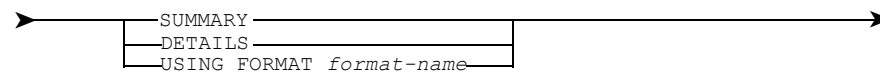


PLOT

To produce a diagrammatic representation of the relations or the records in the Workbench Design Area according to a specified format selection.



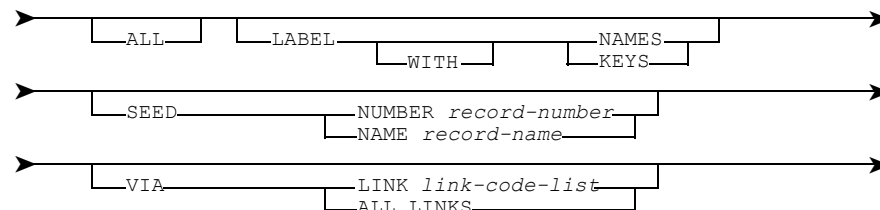
where *format* is:



where:

format-name is the name of a FORMAT member of the repository.

options is:



where:

record-number is the number of a record in the Workbench Design Area.

record-name is a name which has been assigned to a record in the Workbench Design Area, using the NAME command.

link-code-list is a list of link codes which identify the link types to be displayed in the Consolidated Network Plot. If more than one link code is listed, they must be separated by commas and must all be different. Each link code listed must be a number in the range 1 to 12.

POPULATE

To:

- Populate the repository with the results of the preceding PREVIEW IMPORT command.

or

- Generate ENTITY and/or USERVIEW data definitions from some or all of the records and relations in the WorkBench Design Area (WBDA) and enter them as members of the modeling repository.

```
➔ POPULATE [wbta-options] [wbda-options] ; ➔
```

where *wbta-options* are:

```
➔ FROM [BUFFER [USER name] [ROLLBACK [LEVEL [E] [W]]]] ➔
```

where:

name is the name of a USER-MEMBER on the MP-AID.

wbda-options are:

```
➔ [ENTITIES [PREFIX-USERVERIEWS [string] [SUFFIX-USERVERIEWS]]] ➔
  [USERVIEWS]
➔ [AS-MODEL viewset-name] [ALL [NAMES name-list] [NUMBERS range-list]] ➔
➔ [ALPHABETICALLY] [USING FORMAT format] ➔
```

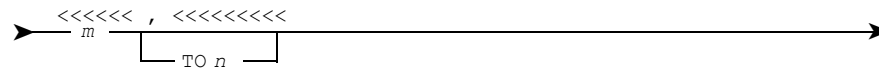
where *string* and *viewset-name* can contain from 1 to 32 characters and must conform to the Manager Products rules for valid member names.

Note:

If a supplementary USERVIEW member is required for an ENTITY being defined, *string* is concatenated, either as a prefix or a suffix, with the name of the ENTITY, to produce the name of the supplementary USERVIEW. If this resulting name contains more than 32 characters, it is reduced.

name-list is a list of names, separated by commas, of relations or records, depending, respectively, on whether USERVIEWS or ENTITIES has been specified in the command.

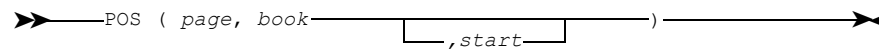
range-list is:



where *m* and *n* are numbers assigned in the Workbench Design Area to relations or records, depending respectively, on whether USERVIEWS or ENTITIES has been specified in the command. If present, *n* must be greater than *m*.

POS Function

To return the position of the first occurrence of one string in another string.



where:

page and *break* are strings.

start is an integer.

POST

See ["MAIL and POST" on page 95](#).

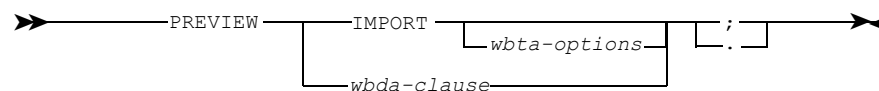
PREVIEW

To

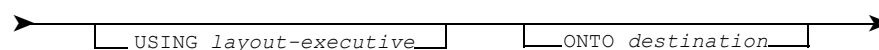
- Generate ADD or REPLACE command and member definition statements from the information on the WorkBench Translation Area (WBTA) documenting external objects.

or

- Generate ENTITY and/or USERVIEW data definitions from some or all of the records and relations in the WorkBench Design Area (WBDA) and display them in the form of repository members.



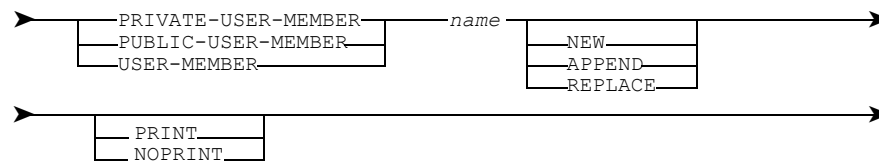
where *wbta-options* are:



where:

layout-executive is the name of an Executive Routine.

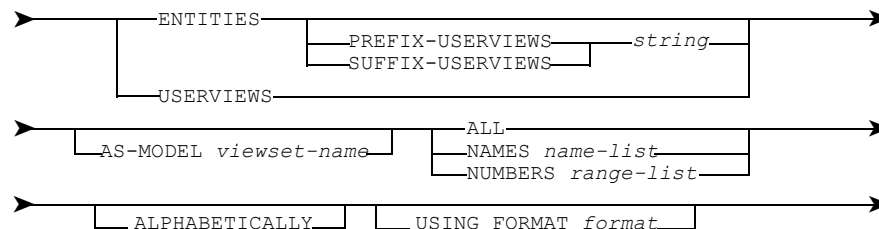
destination is:



where:

name is the name of a USER-MEMBER on the MP-AID.

wbda-clause is:



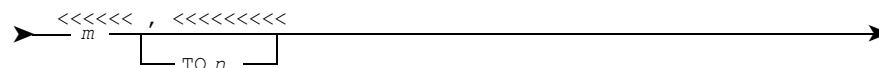
where *string* and *viewset-name* can contain from 1 to 32 alphanumeric characters and must conform to the Manager Products rules for valid member names.

Note:

If a supplementary USERVIEW member is required for an ENTITY being defined, *string* is concatenated, either as a prefix or a suffix, with the name of the ENTITY, to produce the name of the supplementary USERVIEW. If this resulting name contains more than 32 characters, it is truncated.

name-list is a list of names, separated by commas, of relations or records, depending, respectively, on whether USERVIEWS or ENTITIES has been specified in the command.

range-list is:



where *m* and *n* are numbers assigned in the WorkBench Design Area to relations or records, depending, respectively, on whether USERVIEWS or ENTITIES has been specified in the command. If present, *n* must be greater than *m*.

format is the name of a FORMAT member of the Modeling Repository. FORMAT members are used by User Formatted Output functions.

PRINT

To output the source records, or selected parts of the source records, of selected repository members.

<<<<<<<<<< , <<<<<<<<<
PRINT member-name options ;

where:

member-name is the name of a repository member.

options is:

The diagram illustrates the structure of a source file and its encoded representation. The top line, representing the source file, contains three segments: 'LINES' (an integer), 'line-number TO line-number-2', and 'END'. The bottom line, representing the encoded source, contains three segments: 'ALL-STATUSES', 'ENCODED-SOURCE', and 'ALSO'.

where:

integer is a number of lines in the source record.

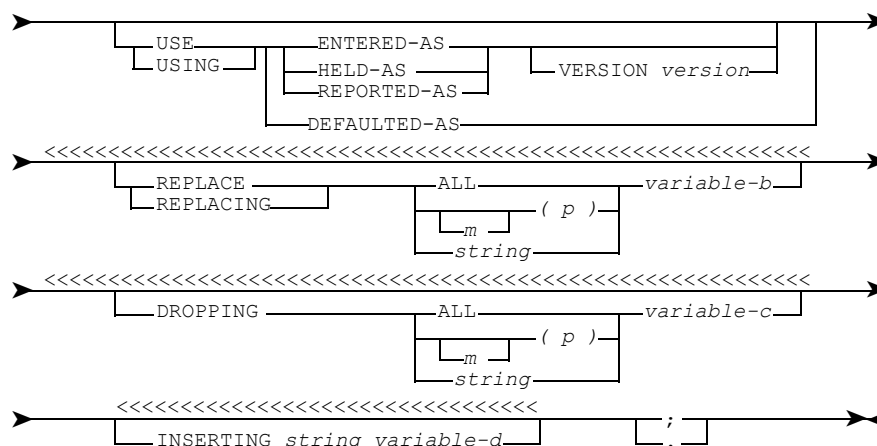
line-number and *line-number-2* are source line numbers.

PRODUCE

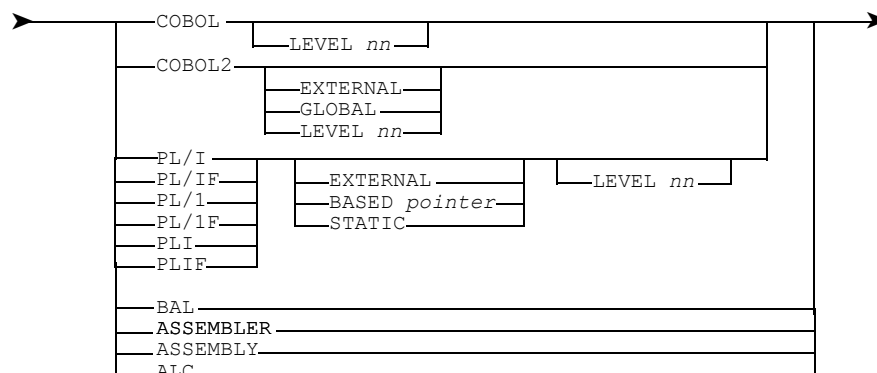
To generate record layouts and/or programming source language statements or database management system language statements from members of the repository.

The diagram illustrates the structure of a COBOL statement, showing the sequence of clauses and their optional sub-clauses. The clauses are represented by horizontal bars, and the sub-clauses are represented by smaller horizontal bars branching off the main clause bars. The sequence of clauses is as follows:

- PRODUCE**: This clause is optional and can be followed by **RECORD-LAYOUTS** and **language**.
- RECORD-LAYOUTS**: This clause is optional and can be followed by **AND** and **FOR**.
- language**: This clause is optional and can be followed by **language**.
- FROM member-name**: This clause is optional and can be followed by **AS library-name**.
- AS library-name**: This clause is optional and can be followed by **PRINT**, **NOPRINT**, and **NO-PRINT**.
- PRINT**, **NOPRINT**, and **NO-PRINT**: These clauses are optional and can be followed by **PRINT**, **NOPRINT**, and **NO-PRINT**.
- ONTO file-name**: This clause is optional and can be followed by **variable-a**.
- NOGENERATION** and **NO-GENERATION**: These clauses are optional and can be followed by **NOGENERATION** and **NO-GENERATION**.
- GIVING output'-form-1**: This clause is optional and can be followed by **OMITTING output'-form-2**.
- OMITTING output'-form-2**: This clause is optional and can be followed by **WITH-ALIAS** and **ALIAS**.
- WITH-ALIAS** and **ALIAS**: These clauses are optional and can be followed by **number** and **alias-type**.



where *language* is:



where:

nn is an assigned integer in the range 1 to 49 inclusive for COBOL, or 1 to 99 inclusive for PL/I, being the initial level of the generated data description.

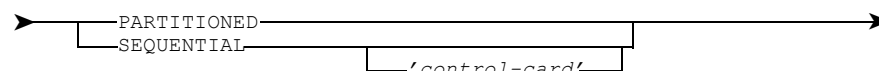
pointer is a PL/I pointer available name indicating the address on which the generated data description is based.

member-name is the name of an encoded repository member from which a record layout and/or a source language dataset description is to be produced. Up to 16 member-names may be declared.

library-name is the name, of not more than 8 characters, to be given to the generated library member in the output dataset. The first character must be alphabetic or @, \$ (or a local currency symbol with the internal code hexadecimal 5B), %, or @.

file-name is the logical file name (ddname or dtfname) used in job control statements to indicate the external dataset name (physical file name) of the dataset to which the generated program is to be written. *file-name* must not be: MPRACWF, MPRDIAG, MPRPOST, MPAID, MPAIDR, MPAIDV or the name of any concatenated MP-AID, the name of the repository, or the repository name with a suffix of: B, C, D, E, F, G, H, I, J, K, L, M, N, R, S, or V.

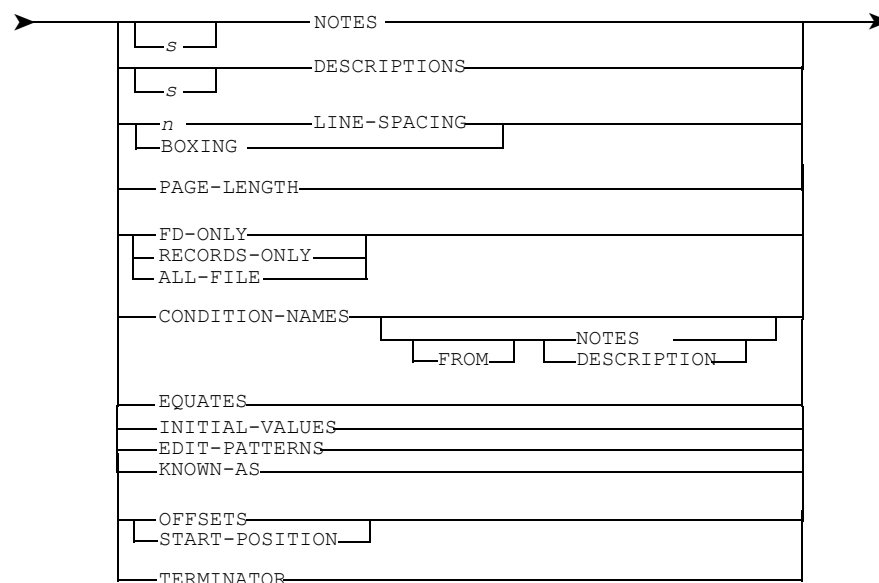
variable-a is:



where:

'control-card' is a character string of up to 72 characters.

output-form-1 is:

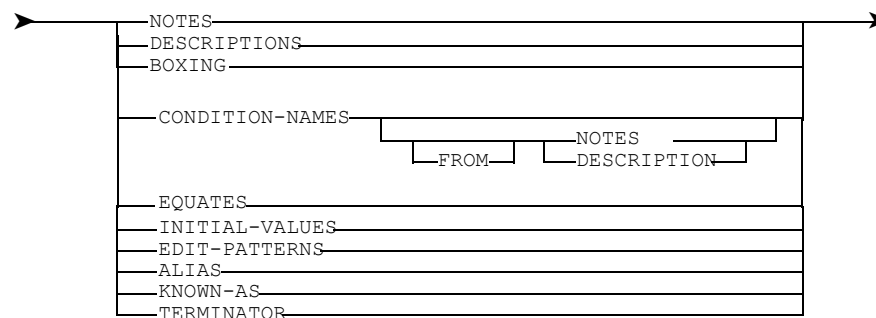


where:

s is an unsigned integer specifying a number of consecutive delimited character strings starting with the first delimited character string of the stated clause.

n is an unsigned integer specifying a number of print lines.

output-form-2 is:



number is an unsigned integer identifying a general alias. The integer must be in the range 1 to g , where g is the number of $ALIAS_n$ keywords of the DALIAS macro that have been implemented with empty values. The maximum possible value of g is 16.

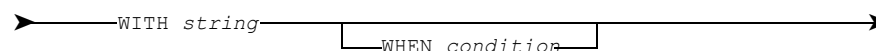
alias-type is a keyword from the alias-type keyword list of the repository.

version is an unsigned integer in the range 1 to 15.

m and p are unsigned integers in the range 1 to 96, specifying a generated data name, starting at character position m of the name and including p characters. If m is omitted a value of 1 is defaulted. The sum of m and p must not exceed 97. A space or spaces must separate m and (p) .

string is a character string of not more than 32 printable characters. A space (hexadecimal 40) is considered to be a printable character.

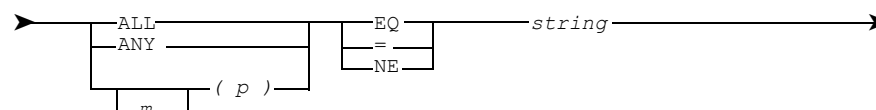
variable-b is:



where:

string is as defined above.

condition is:



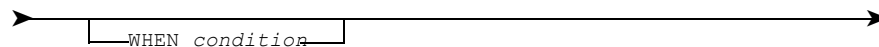
where:

m is as defined above.

p is as defined above.

string is as defined above.

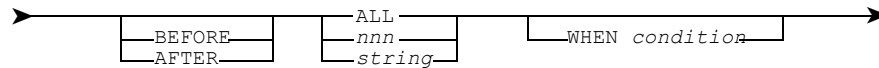
variable-c is:



where:

condition is as defined above.

variable-d is:



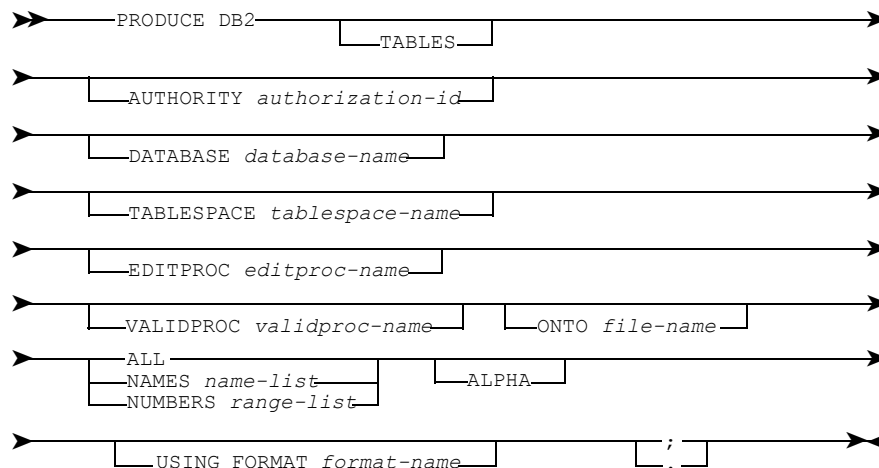
where:

nnn is an unsigned integer in the range 1 to 96, specifying a character position in a generated data name condition is as defined above.

string is as defined above.

PRODUCE

To generate a set of CREATE TABLE commands for use in DB2.



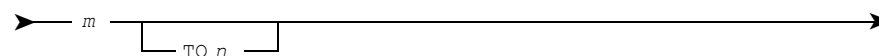
where:

authorization-id, *database-name*, *tablespace-name*, *editproc-name*, and *validproc-name* are undelimited names, no more than eight characters in length. They can consist of any letters of the alphabet, the numerals zero to nine, @ symbols, # symbols, and the local currency symbol with the internal code of hexadecimal 5B. The first character of the name must not be a numeral.

file-name is the name of a file in which the generated SQL CREATE TABLE statements are to be stored.

name-list is a list of relation names, separated by commas.

range-list is a list of numeric ranges, separated by commas, each of the form:



where *m* and *n* are numbers assigned to relations in the Workbench Design Area. If present, *n* must be greater than *m*.

format-name is the name of a FORMAT member of the Modeling Repository.

PRODUCE DLI DATABASE DESCRIPTION

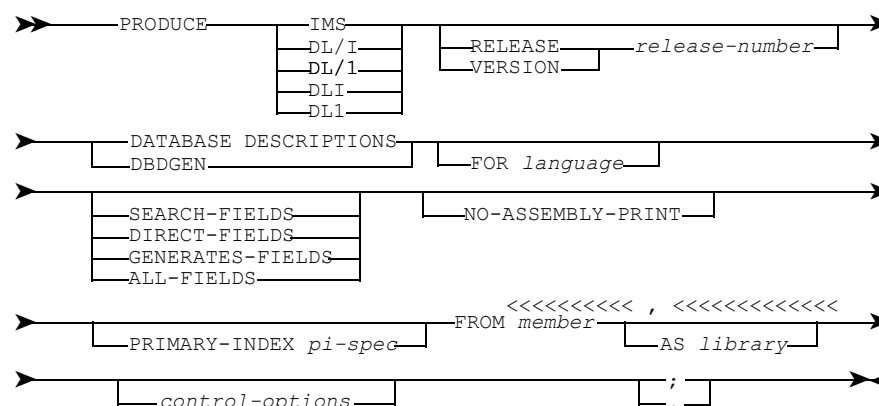
See ["PRODUCE IMS DATABASE DESCRIPTION" on page 122](#).

PRODUCE DLI PROGRAM SPECIFICATION BLOCK

See ["PRODUCE IMS PROGRAM SPECIFICATION BLOCK" on page 123](#).

PRODUCE IMS DATABASE DESCRIPTION

To produce IMS (DLI) Database Description Control Statements which can subsequently be used as input for an IMS (DLI) DBD Generation.

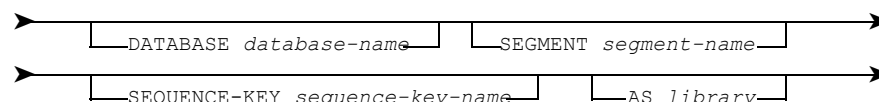


where:

release-number is 1.2, 1.3, 2, 2.0, 2.1, 2.2, 3, 3.0, 3.1, 4.0, or 4.1.

language is COBOL, COBOL2, COBOL-2, COBOLII, COBOL-II, PL/I, PL/1, PLI, PL1, PL/IF, PL/1F, or PLIF.

pi-spec is:



where:

database-name, *segment-name*, and *sequence-key-name* are valid IMS (DL/I) names.

library is string of up to 16 characters. The first character must be: #, alphabetic, local currency symbol (internal code hexadecimal 5B), %, or @.

member is an IMS-DATABASE or DL/I-DATABASE member.

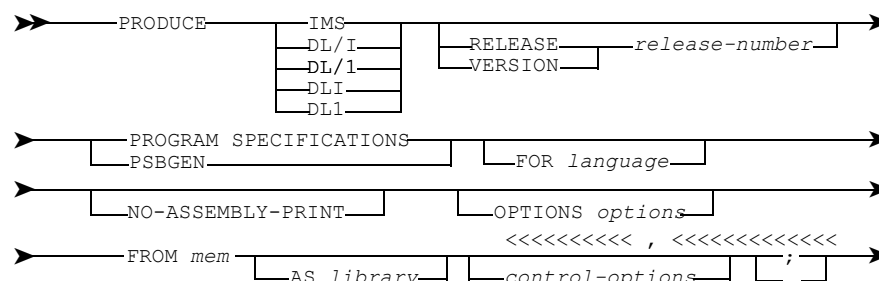
library is as defined above.

control-options is a series of optional clauses that are defined in the documentation describing the Source Language Generation facility, except that:

- The USE or USING clause defined there is excluded.
- Only the KNOWN-AS option is valid in the GIVING clause.
- Only the KNOWN-AS/ALIAS options are valid in the OMITTING clause.
- If you specify NO-GENERATION or NO-PRINT, no processing occurs.

PRODUCE IMS PROGRAM SPECIFICATION BLOCK

To produce IMS (DLI) Program Specification Block Control Statements which can subsequently be used as input for an IMS (DLI) PSB Generation.

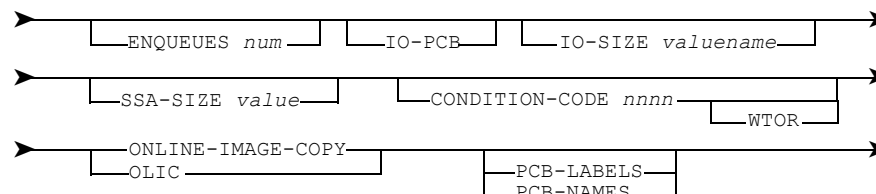


where:

release-number is 1.2, 1.3, 2, 2.0, 2.1, 2.2, 3, 3.0, 3.1, 4.0, or 4.1.

language is COBOL, COBOL2, COBOL-2, COBOLII, COBOL-II, PL/I, PL/1, PLI, PL1, PL/IF, PL/1F, or PLIF.

options are:



where:

num is an unsigned integer.

value is an unsigned integer not greater than 256000.

nnn is an unsigned integer in the range 0 to 4095.

mem is the name of a SYSTEM, PROGRAM, or MODULE member.

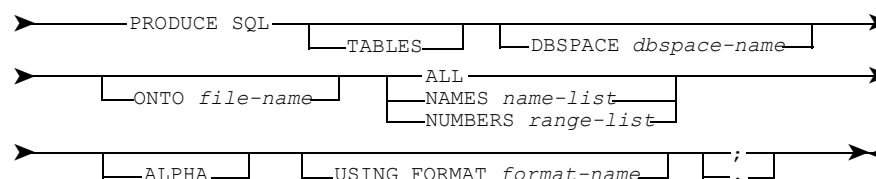
library is a string of up to 16 characters. The first character must be: #, alphabetic, local currency symbol (internal code hexadecimal 5B), %, or @.

control-options is a series of optional clauses that are defined in the documentation describing the Source Language Generation facility, except that:

- The USE or USING clause defined there is excluded.
- Only the KNOWN-AS option is valid in the GIVING clause.
- Only the KNOWN-AS/ALIAS options are valid in the OMITTING clause.
- If you specify NO-GENERATION or NO-PRINT, no processing occurs.

PRODUCE SQL

To generate a set of CREATE TABLE commands for use in SQL/DS.



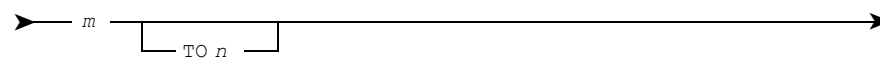
where:

dbspace-name is the name of the dbspace and must conform to the SQL/DS rules for naming a dbspace.

file-name is the name of a file in which the generated SQL CREATE TABLE statements are to be stored.

name-list is a list of relation names, separated by commas.

range-list is a list of numeric ranges, separated by commas, each of the form:

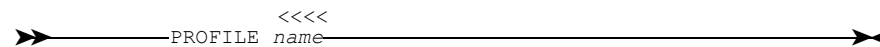


where *m* and *n* are numbers assigned to relations in the Workbench Design Area. If present, *n* must be greater than *m*.

format-name is the name of a FORMAT member of the repository.

PROFILE

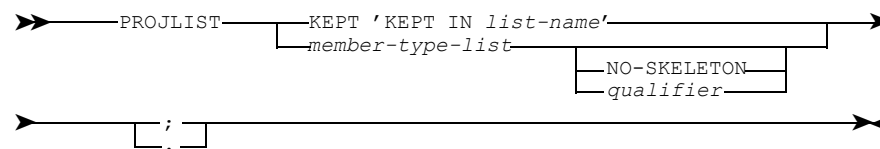
To declare profile variables.



where *name* is any valid user-defined variable name.

PROJLIST

To list a selection of the members belonging to the current project.

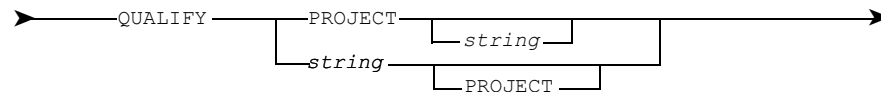


where:

list-name is the name of a KEPT-DATA list you have created.

member-type-list is a list of any of the member types available to your repository.

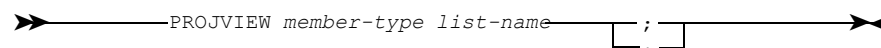
qualifier is:



where *string* is any character string.

PROJVIEW

To create a KEPT-DATA list containing all the members of a specified member type that belongs to the current project.



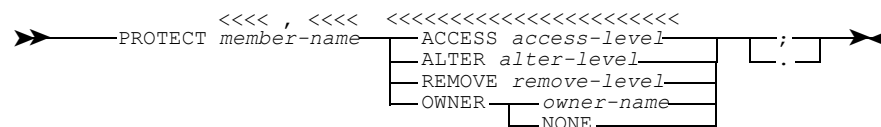
where:

member-type is any member-type available in your repository.

list-name is the name of the KEPT-DATA list.

PROTECT

To protect members of the repository against unauthorized access, alteration, or removal.



where:

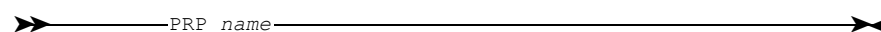
member-name is the name of a repository member.

access-level, *alter-level*, and *remove-level* are integers in the range 0 to 254, specifying the security levels.

owner-name is a string of up to 32 printable characters, being the name of an owner as previously established by the Controller.

PRP

To generate a valid name for a member from the standard abbreviation table.



where *name* is the name of a member in your repository.

PRPD

To display the entries in the standard abbreviation table.

➤ —PRPD *name* —➤

PTIME Function

To return the time in *hh.mm.ss.ttt* format (*ttt* is thousandths of a second).

➤ —PTIME () —➤

PULL

To allow an environment whose configuration has been temporarily changed to revert to its original configuration.

➤ —PULL — [] ; [] —➤

PUR and PURGE

To remove a repository member to which there are no references.

Line Command:

➤ —PUR —➤

Primary Command:

➤ —PURGE *member-name* — [] ; [] —➤

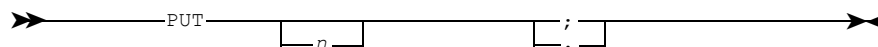
PUSH

To store information about your current working environment so that you can work in a differently configured environment before reverting to your original environment.

➤ —PUSH — [] ; [] —➤

PUT

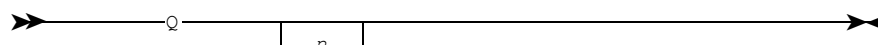
To copy lines to the Scratchpad from the current line onward. The current contents of the Scratchpad are overwritten. Lines copied are not removed from the current buffer.



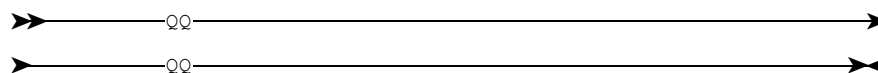
where n is the number of lines to be copied.

Q and QQ Line Commands

To add a line or block of lines to the existing Scratchpad.

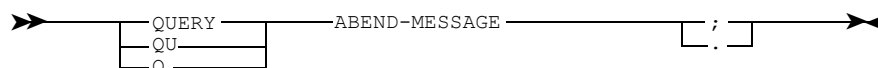


where n is an integer in the range 1 to 9999.



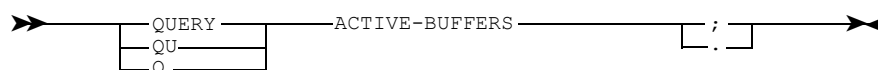
QUERY ABEND-MESSAGE

To find out whether an abend message is set.



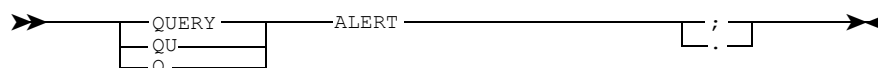
QUERY ACTIVE-BUFFERS

To find out which Command Mode, Update, Edit, and Lookaside Buffers you currently have open, and, in the case of Edit and Update Buffers, which members are contained in the buffers.



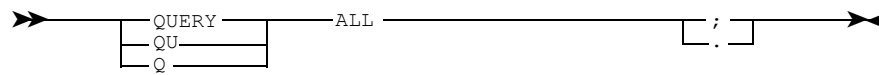
QUERY ALERT

To find out whether the audible alert is set on or off.

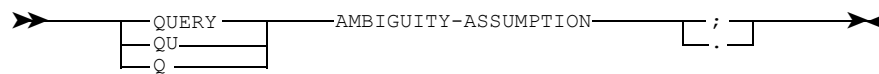


QUERY ALL

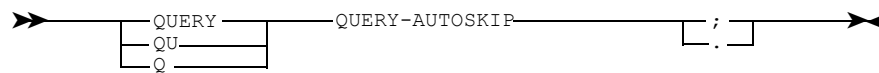
To obtain a complete description of the current user environment.

**QUERY AMBIGUITY-ASSUMPTION**

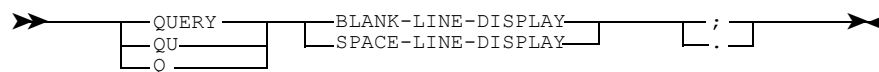
To find out whether the command ambiguity assumption feature is set on or off.

**QUERY AUTOSKIP**

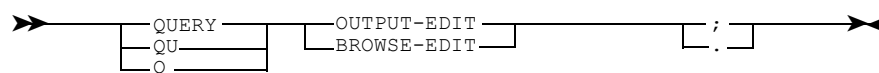
To find out whether the Autoskip feature is set on or off.

**QUERY BLANK-LINE-DISPLAY and QUERY-SPACE-LINE-DISPLAY**

To find out whether the Blank Line Display capability is set on or off.

**QUERY BROWSE-EDIT and QUERY OUTPUT-EDIT**

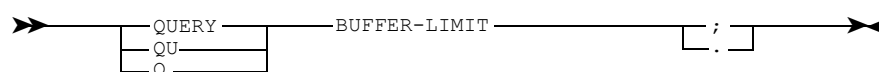
To check whether the SET BROWSE-EDIT or OUTPUT-EDIT facility is set on or off.

**QUERY BROWSE-RETENTION**

See ["QUERY LOOKASIDE-RETENTION and QUERY BROWSE-RETENTION" on page 137.](#)

QUERY BUFFER-LIMIT

To find out the limit on the number of buffers available.

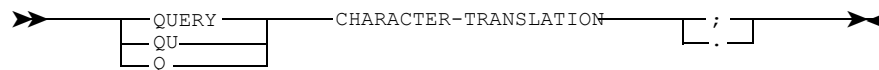


QUERY CEI

See ["QUERY CORPORATE-EXECUTIVE-INDEX" on page 130.](#)

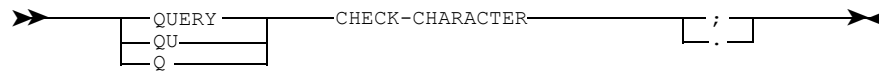
QUERY CHARACTER-TRANSLATION

To display the character translation implemented via the SET CHARACTER TRANSLATION command.



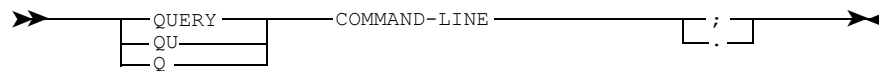
QUERY CHECK-CHARACTER

To find out the current setting of the SET CHECK-CHARACTER command.



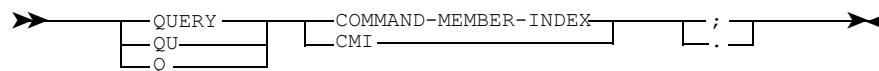
QUERY COMMAND-LINE

To find out the current position of the Command Area.



QUERY COMMAND-MEMBER-INDEX

To obtain information about the size and contents of the run-time COMMAND-MEMBER index.

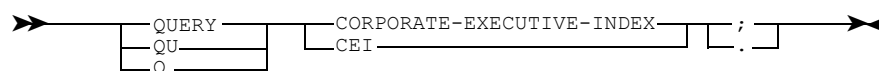


QUERY COMMAND-RETENTION

See ["QUERY EXECUTIVE-RETENTION and QUERY COMMAND-RETENTION" on page 133.](#)

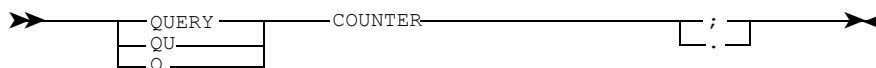
QUERY CORPORATE-EXECUTIVE-INDEX

To obtain information about the size and contents of the run-time Corporate Executive index.

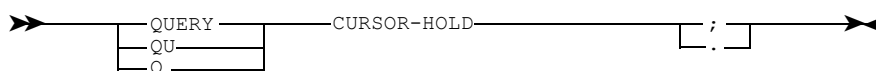


QUERY COUNTER

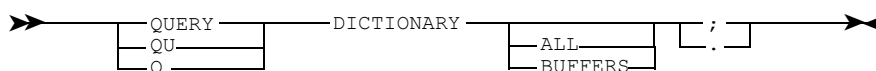
To find out the increment at which the Line Counter (displaying the number of lines accumulating in the current buffer) changes.

**QUERY CURSOR-HOLD**

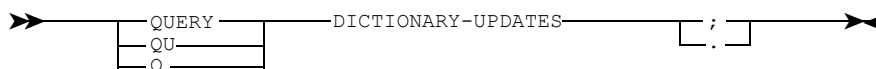
To find out the current setting of the Cursor Hold capability.

**QUERY DICTIONARY**

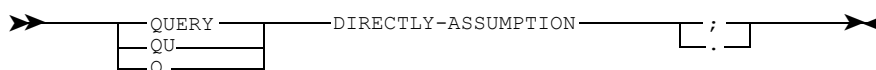
To find out details of the repository, status, repository dataset utilization, and the UDS table name.

**QUERY DICTIONARY-UPDATES**

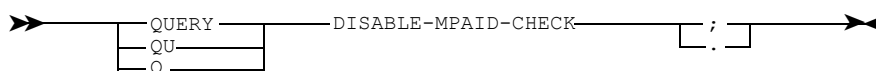
To find out if repository updates are permitted.

**QUERY DIRECTLY-ASSUMPTION**

To find out whether the SET DIRECTLY command is set on or off.

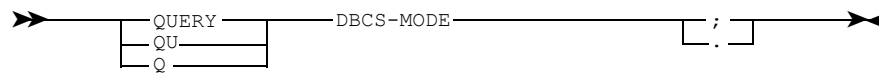
**QUERY DISABLED-MPAID-CHECK**

To enable the current setting of the SET DISABLED-MPAID-CHECK command to be found. SET SIDABLED-MPAID-CHECK prevents the MP-AID being checked whenever a command is entered, to see if it has been disabled by the System Administrator since the execution of the last command.



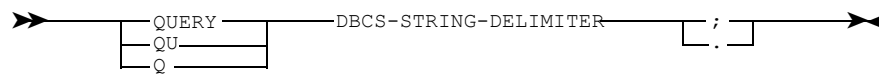
QUERY DBCS-MODE

To find out the current setting of the double byte character set mode.



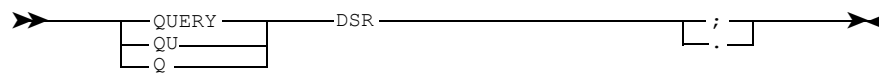
QUERY DBCS-STRING-DELIMITER

To find out the current setting of the double byte character set string delimiter.



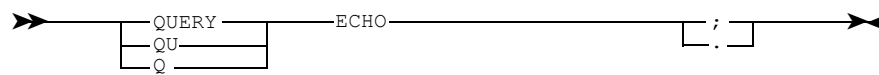
QUERY DSR

To find out if the Data Modeling and Design environment is active.



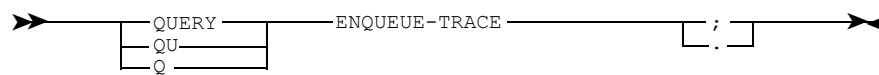
QUERY ECHO

To find out whether the Command Echo capability is set on or off.



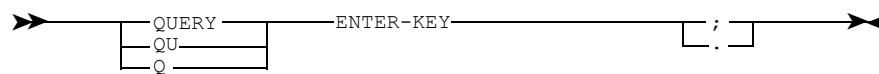
QUERY ENQUEUE-TRACE

To determine if the Enqueue Trace capability is active.



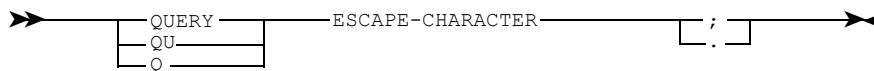
QUERY ENTER-KEY

To find out the setting of the ENTER-KEY.

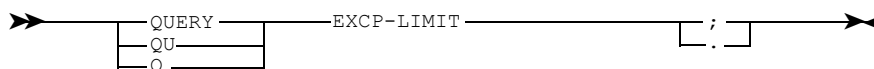


QUERY ESCAPE-CHARACTER

To find out which character is being used as the escape character.

**QUERY EXCP-LIMIT**

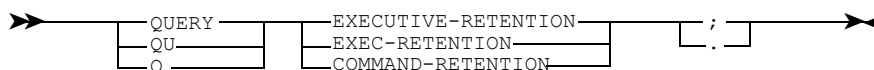
To find out the maximum number of Input/Output operations permitted during the execution of a single command.

**QUERY EXCP-MONITOR**

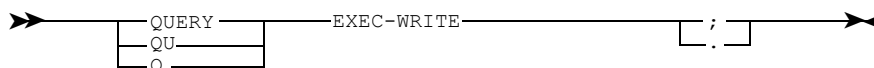
To find out whether the Excp Monitor capability is set to an integer value or is set off.

**QUERY EXECUTIVE-RETENTION and QUERY COMMAND-RETENTION**

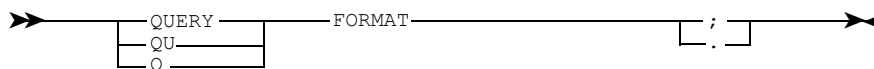
To find out whether Executive Routines which include the RETAIN directive are retained in virtual storage after their initial execution.

**QUERY EXEC-WRITE**

To find out the current setting of the SET EXEC-WRITE command.

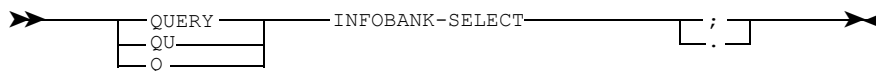
**QUERY FORMAT**

To find out whether any of the preferred set of alternative formats or any user-defined format will be applied to output from the REPORT or BULK REPORT commands.

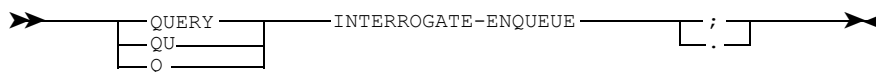


QUERY INFOBANK-SELECT

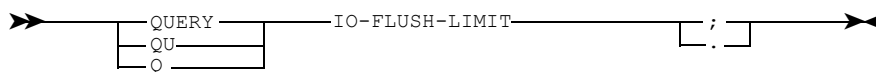
To find out the current setting of the InfoBank SELECT keyword display capability.

**QUERY INTERROGATE-ENQUEUE**

To find out whether INTERROGATE-ENQUEUE is set ON or OFF.

**QUERY IO-FLUSH-LIMIT**

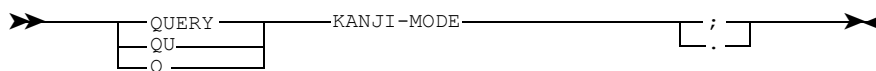
To find out whether a limit has been put on the number of permissible I/O flushes and if so, what the limit value is.

**QUERY IPI**

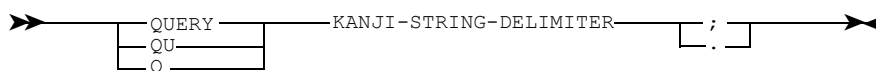
See ["QUERY INFOBANK-PANEL-INDEX" on page 134](#).

QUERY KANJI-MODE

To find out which Kanji Mode is set.

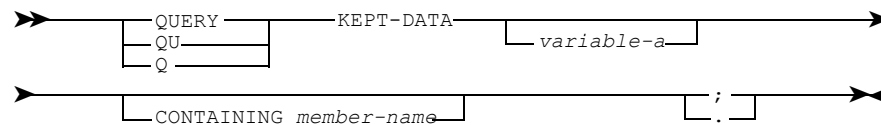
**QUERY KANJI-STRING-DELIMITER**

To find out which special string delimiter has been set for use in a Kanji environment.

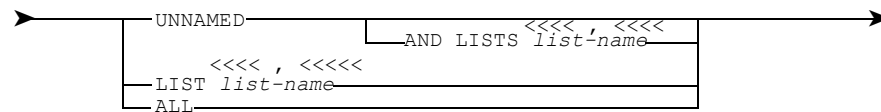


QUERY KEPT-DATA

To find out what KEPT-DATA lists, if any, you have created.



where *variable-a* is:



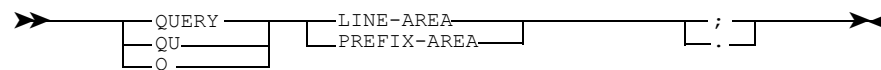
where:

list-name is the name of a KEPT-DATA list.

member-name is the name of a specific member.

QUERY LINE-AREA and QUERY PREFIX-AREA

To find out the current position of the Line Command Area.



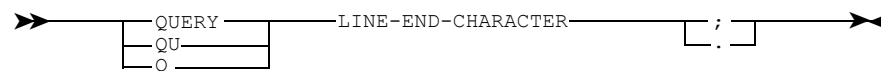
QUERY LINE-COMMANDS and QUERY PREFIX-COMMANDS

To find out whether any Line Commands have been renamed.



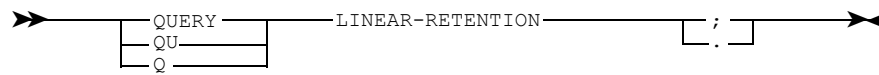
QUERY LINE-END-CHARACTER

To find the current Line End Character.

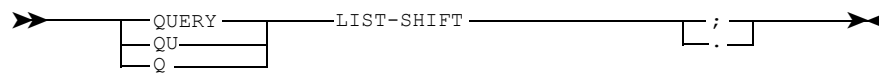


QUERY LINEAR-RETENTION

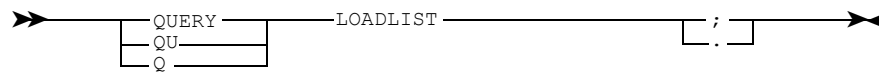
To find out the current setting of the Linear Retention capability.

**QUERY LIST-SHIFT**

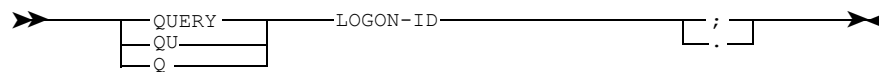
To find out whether the List-shift feature is operative.

**QUERY LOADLIST**

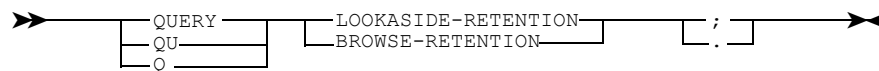
To display information about dynamically loaded Manager Products components.

**QUERY LOGON-ID**

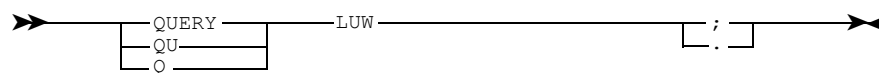
To find out the Logon Identifier with which you are logged on.

**QUERY LOOKASIDE-RETENTION and QUERY BROWSE-RETENTION**

To find out whether the Lookaside Retention feature is set on or off.

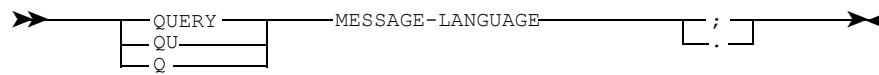
**QUERY LUW**

To find out if there is an active Logical Unit of Work (LUW).



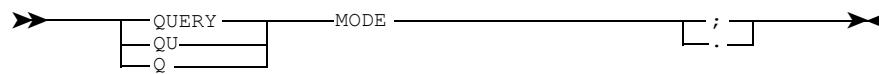
QUERY MESSAGE-LANGUAGE

To find out in which language messages will be displayed.



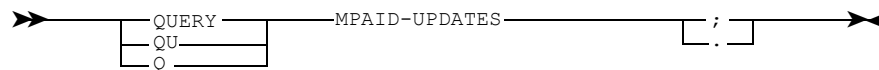
QUERY MODE

To determine the environment to which a cross-product command is to be routed.



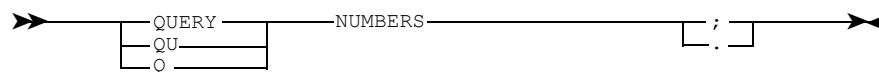
QUERY MPAID-UPDATES

To find out if MP-AID updates are permitted.



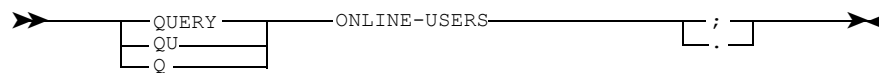
QUERY NUMBERS

To find out whether the Numbers feature for source lines of repository members is set on or off.



QUERY ONLINE-USERS

To find out which users are logged on to Manager Products, excluding batch users.

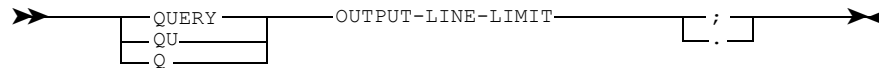


QUERY OUTPUT-EDIT

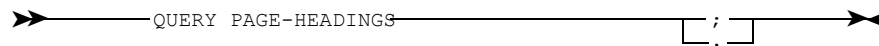
See ["QUERY BROWSE-EDIT and QUERY OUTPUT-EDIT" on page 129.](#)

QUERY OUTPUT-LINE-LIMIT

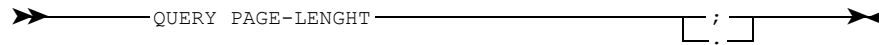
To find out the maximum number of lines that can be held in the Command Mode Buffer and each Lookaside Buffer.

**QUERY PAGE-HEADINGS**

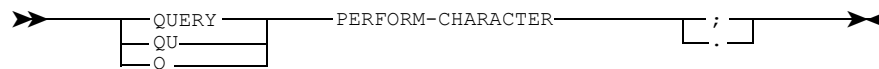
To find out the current setting of the SET PAGE-HEADINGS command.

**QUERY PAGE-LENGTH**

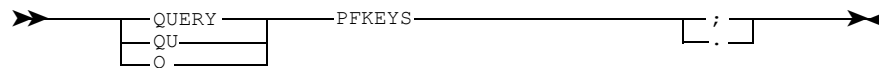
To find out the current setting of the SET PAGE-LENGTH command.

**QUERY PERFORM-CHARACTER**

To find out which character is specified as the Perform Character for PERFORM commands and COMMAND-STREAM members.

**QUERY PFKEYS**

To find out the settings of the PF keys.

**QUERY PREFIX-AREA**

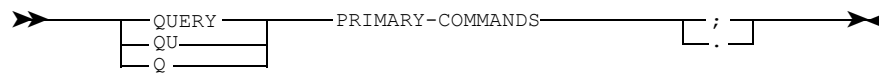
See ["QUERY LINE-AREA and QUERY PREFIX-AREA" on page 136](#).

QUERY PREFIX-COMMANDS

See ["QUERY LINE-COMMANDS and QUERY PREFIX-COMMANDS" on page 136](#).

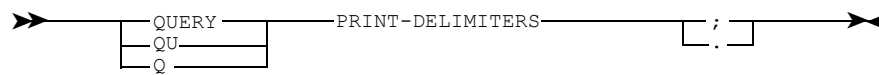
QUERY PRIMARY-COMMANDS

To find out whether any Primary Commands have been renamed or disabled.



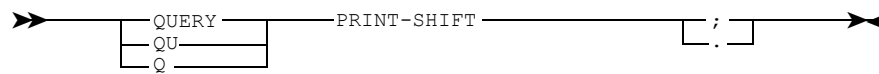
QUERY PRINT-DELIMITERS

To find out whether the PRINT OF . . ./END of PRINT delimiters for prints of repository and MP-AID members are set on or off.



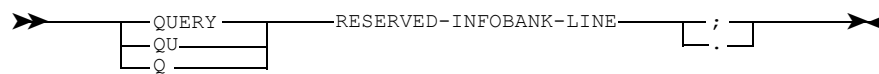
QUERY PRINT-SHIFT

To find out the current setting of the Print-Shift feature.



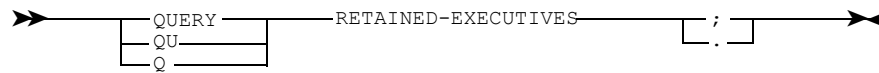
QUERY RESERVED-INFOBANK-LINE

To find out the text currently specified for the Reserved InfoBank Line.



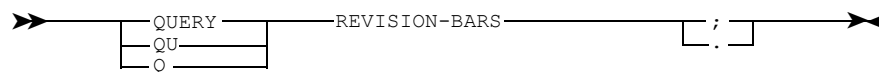
QUERY RETAINED-EXECUTIVES

To obtain information about Executive Routines that are retained in virtual storage.



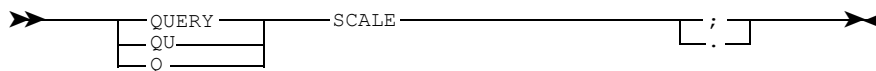
QUERY REVISION-BARS

To find out the current setting of the Revision Bar capability.

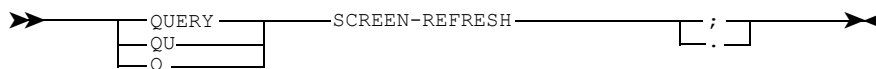


QUERY SCALE

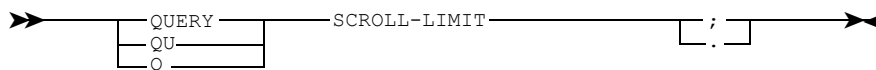
To find out whether the Scale feature of the General Purpose Screen is set on or off.

**QUERY SCREEN-REFRESH**

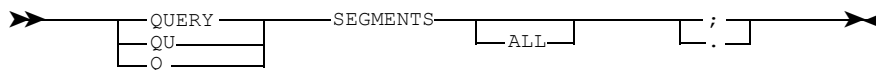
To find out the current setting of the Screen Refresh capability.

**QUERY SCROLL-LIMIT**

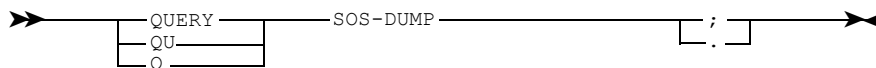
To find out whether the Scroll Limit capability (retention of lines in Command Mode Buffer) is set on or off, or the value to which it is set.

**QUERY SEGMENTS**

To display information about the usage of Manager Products program code segments.

**QUERY SOS-DUMP**

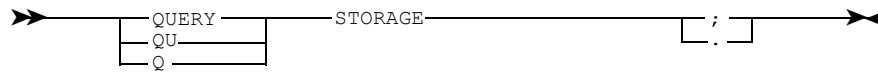
To find the current setting of the SOS-DUMP capability.

**QUERY SPACE-LINE-DISPLAY**

See ["QUERY BLANK-LINE-DISPLAY and QUERY-SPACE-LINE-DISPLAY" on page 129.](#)

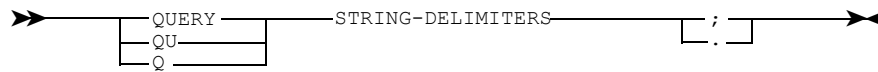
QUERY STORAGE

To find the current Virtual Storage usage (synonymous with QUERY VIRTUAL).



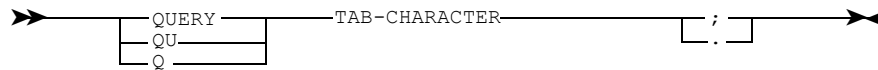
QUERY STRING-DELIMITER

To find which characters may be used to delimit strings.



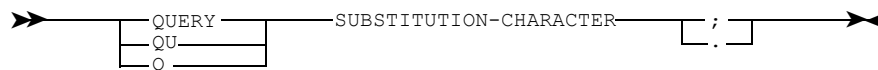
QUERY TAB-CHARACTER

To find which character has been specified as a tab character.



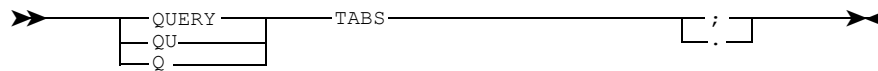
QUERY SUBSTITUTION-CHARACTER

To find the substitution character in use with the substitutable.



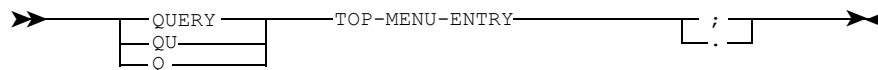
QUERY TABS

To find the current tab position settings.



QUERY TOP-MENU-ENTRY

To find the name of the panel accessed by a SELECT TOP-MENU command. (The ASG-supplied default is HELP000000.)



QUERY TRACE

To display Procedures Language and Manager Products trace settings which are currently in force. Trace facilities are invoked by the SET TRACE command.

```

>> QUERY TRACE _____ [ ; ] >>

```

QUERY TRANSLATION

To display any default Translation Rules previously specified for the transfer of repository members to other repositories or directories.

```

>> QUERY _____ TRANSLATION _____ >>
    QU _____
    Q _____

> OTHER-DICTIONARY _____
  IDMS _____
  IDD _____
    TRANSLATION-RULES _____
    TR-RULES _____
    SIGNON _____
    OPTIONS _____
    TRANSLATION-RULES _____
    TR-RULE _____

> _____ [ ; ] >>

```

QUERY UMI

See ["QUERY USER-MEMBER-INDEX" on page 144](#).

QUERY UPDATE-OUTPUT

To find out in which circumstances the text of a repository member is displayed when the member is filed.

```

>> QUERY _____ UPDATE-OUTPUT _____ [ ; ] >>
    QU _____
    Q _____

```

QUERY UPPER-CASE

To find out whether the Upper Case capability (conversion of lowercase alphabetic characters) has been set on or off.

```

>> QUERY _____ UPPER-CASE _____ [ ; ] >>
    QU _____
    Q _____

```

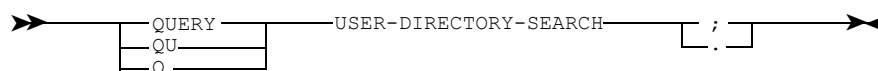
QUERY USER-BLOCKS

To find out the number of blocks on the MP-AID occupied by USER-MEMBERS owned by your Logon Identifier.



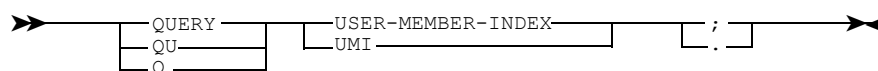
QUERY USER-DIRECTORY-SEARCH

To find out the current setting of the USER-DIRECTORY-SEARCH command.



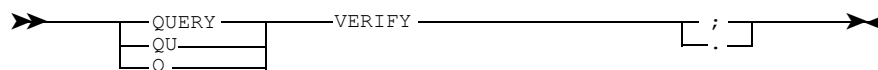
QUERY USER-MEMBER-INDEX

To obtain information about the size and contents of the run-time user member index.



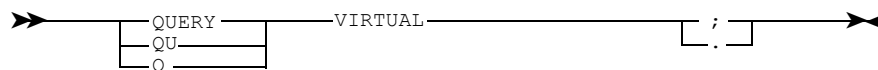
QUERY VERIFY

To find out the current setting of the SET VERIFY command.



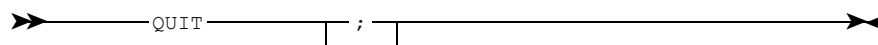
QUERY VIRTUAL

To find out the current Virtual Storage usage (synonymous with QUERY STORAGE).



QUIT

To close and delete a Lookaside Buffer, or an Update or Edit Buffer which has not been amended.



R Line Command

To duplicate a line.

```

>>-----R-----[ n ]----->>

```

where n is an integer.

RADD

To specify that you want a proposed member documenting an external object to be added to the repository.

Line Command:

```

>>-----RADD----->>

```

Primary Command:

```

>>-----RADD n-----[ ; ]----->>

```

where n is a proposed member's identification number in a Reconciliation Report.

READ

To read the contents of a CMS file.

```

>>-----READ-----[ fn ]-----[ ; ]----->>
                |
                |-----[ ft ]-----
                |
                |-----[ fm ]-----

```

where fn , ft , and fm are the file name, file type, and file mode of a CMS file.

To read from an MVS dataset.

```

>>-----READ-----[ ddname ]-----[ ; ]----->>
                |
                |-----[ member-name ]-----
                |
                |-----[ DIRECTORY ]-----

```

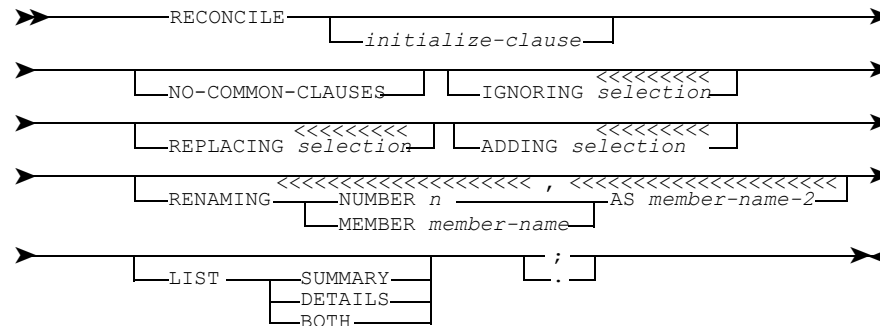
where:

ddname is the ddname specified in a previously issued TSO ALLOCATE command and is mandatory.

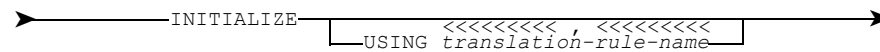
member-name is mandatory and is either the name of a PDS member or the keyword DIRECTORY if the directory is to be read. The keyword DIRECTORY may not be truncated.

RECONCILE

To generate proposed members from the information about external objects held on the WorkBench Translation Area and reconcile the proposed members with the current contents of the repository.



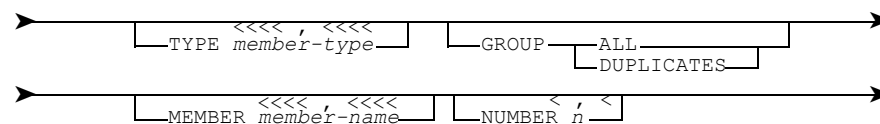
where *initialize-clause* is:



where:

translation-rule-name is the name of an Executive Routine.

selection is:



where:

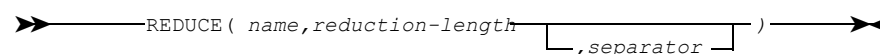
member-type is the member type of a proposed member.

member-name is the name of a proposed member.

n is a proposed member's identification number in a Reconciliation Report.

REDUCE Function

To output a reduced version of a name (typically a member name) according to specific rules.



where:

name is a string of up to 80 characters.

reduction-length is an integer between 1 and 50.

separator is a character.

REF and REFERENCES

To list all repository members to which a specified member refers directly

Line Command:

```
➤➤-----REF----->>
```

Primary Command:

```
➤➤-----REFERENCES member-name-----[ ] ; [ ]>>
```

REFA

To list all repository members to which a specified member refers directly or indirectly.

Line Command:

```
➤➤-----REFA----->>
```

Primary Command:

```
➤➤-----REFA member-name-----[ ] ; [ ]>>
```

REINSTATE

To remove an unverified source record, reinstating a member to the condition in which it was last successfully encoded.

```
➤➤-----REINSTATE <<<< , <<<< member-name-----[ ] ; [ ]>>
```

where *member-name* is the name of a repository member.

member-type is the interrogate keyword for any member type or collective member type available in your repository.

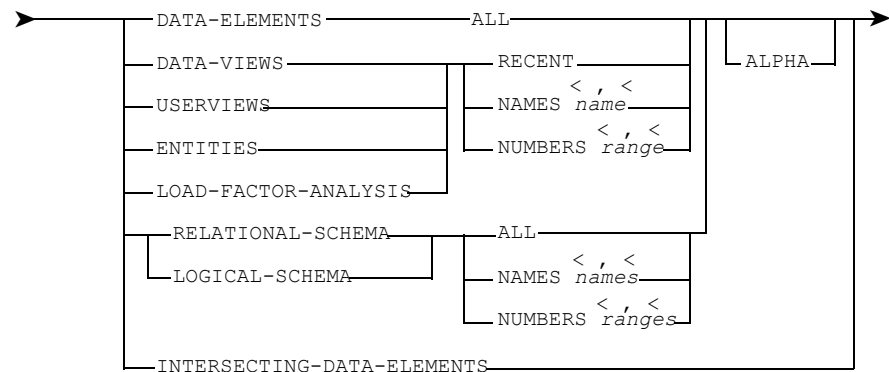
Use the SHOW UDS command to find out which member types are available in your repository.

REPORT Data Modeling and Design Command

To produce a report on a selected category of data in the WorkBench Design Area.

➤➤ ——— REPORT *report-selection* ——— ➤➤

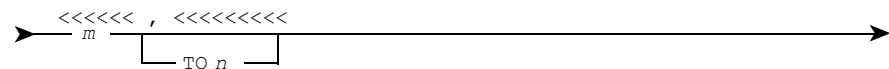
where *report-selection* is:



where:

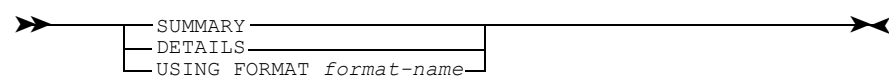
names is a list of names, separated by commas, of data elements, USERVIEWS, data-views, ENTITIES, relations, as indicated by the preceding report category keyword.

ranges is a list of numeric ranges separated by commas, each of the form:



where *m* and *n* are numbers assigned in the WBDA to data elements, USERVIEWS, data-views, ENTITIES, relations, as indicated by the preceding report category keyword. If present, *n* must be greater than *m*.

format-selection is:



where *format-name* is the name of a FORMAT member of the modeling repository.

REPSTR Function

To replace occurrences of one string by another.

➡ REPSTR(*str1* *str2* , *str3*) ➡

where *str1*, *str2*, and *str3* are strings.

RESERVE Executive Command

To define the start of a Logical Unit of Work (LUW).

```

sequenceDiagram
    participant Client
    participant Server
    Client->>Server: RESERVE
    Server->>Client: DICTONARY
    Client->>Server: UPDATE
    Server->>Client: COMMIT
    Server->>Client: ROLLBACK
    Server->>Client: SPEED
    Client->>Server: READ-ONLY
    Server->>Client: SHARED
    Client->>Server: MP-AID
    Server->>Client: UPDATE
    Client->>Server: EXCLUSIVE
    Server->>Client: READ-ONLY
    Client->>Server: SHARED
    Server->>Client: SHARED
    Client->>Server: SHARED
    Server->>Client: SHARED
  
```

RESTART

To enable a user to log off from a Manager Products session and be presented with an ASG Logon Panel to allow logon at a later time.

A horizontal timeline with a double arrow at the left end. The word "RESTART" is written on the line. To the right of "RESTART", there is a small rectangular box containing a semicolon ";". A vertical line segment extends downwards from the bottom of this box. The timeline ends with a double arrow at the right end.

RETAIN Directive

To cause an Executive Routine to be retained in virtual storage.

RETAINS

RETRACE

To display or list the InfoBank panels you have previously seen.

RETRACE

PANEL

MENU

DISPLAY

RETURN

To enable a user to switch from a virtual machine environment to a Manager Products environment.

➤—RETURN—➤

RETURN Directive

To cause control to pass back to the most recent active CALL.

➤—RETURN—➤

REVERSE Function

To return a string, reversed end-to-end.

➤—RETURN(*string*)—➤

where *string* is a string.

REVERT

To remove a member from the current status to allow its definition in a base status to be seen.

➤—REVERT ^{<<<< , <<<<}
member-name [;]—➤

where *member-name* is the name of member in a Manager Products repository.

RFILE

To replace the source record of an existing member with the contents of the Update Buffer, and attempt to encode it.

➤—RFILE [*member-name*] [;]—➤

where *member-name* is the name of a repository member.

RIGHT Function

To return a sub-string from the end of the given string.

➤—RIGHT(*string*, *length* [, *pad*])—➤

RIGHT

To display text to the right of the text currently displayed.

➤ — RIGHT ; — ➤

RIGN

To specify that you do not want a proposed member documenting an external object to be entered in the repository.

Line Command:

➤ — RIGN — ➤

Primary Command:

➤ — RIGN *n* ; — ➤

where *n* is a proposed member's identification number in a Reconciliation Report.

ROOT Function

To return the *n*th root of a number, rounded up or down, if necessary, to the nearest integer.

➤ — ROOT (*number* *n*th , *round*) — ➤

where:

number is a value between -1073741824 and +1073741824.

n is an integer between 1 and 15.

round is d or D or u or U.

RREN

To rename a proposed member documenting an external object.

Line command:

➤ — RREN — ➤

Primary command:

➤➤ —RREN *n* — [] ; [] —➤➤

where *n* is a proposed member's identification number in a Reconciliation Report.

RREP

To specify that you want a proposed member documenting an external object to replace an existing repository member.

Line command:

➤➤ —RREP —➤➤

Primary command:

➤➤ —RREP *n* — [] ; [] —➤➤

where *n* is a proposed member's identification number in a Reconciliation Report.

RUN

To execute permitted Manager Products commands from the Edit Buffer.

➤➤ —RUN — [] ; [] —➤➤

S and SS Line Commands

To remove lines form an Edit or Update Buffer and put them into the Scratchpad. The current contents of the Scratchpad are NOT overwritten.

➤➤ —S — [] *n* [] —➤➤

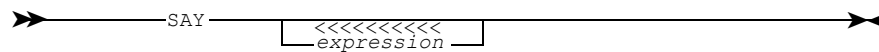
where *n* is an integer in the range 1 to 9999.

➤➤ —SS —➤➤

➤ —SS —➤➤

SAY Directive

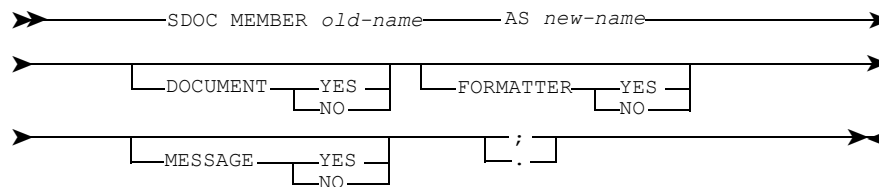
To output text to the primary/secondary output only.



where *expression* is an expression subject to Full Evaluation.

SDOC

To assemble a document, optionally filing a copy of the assembled document in the repository.



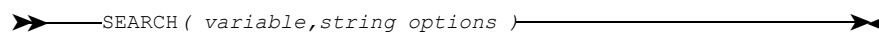
where:

old-name is the name of a DOCUMENT member.

new-name is the proposed name for the formatted document.

SEARCH Function

To search an array for a given string.

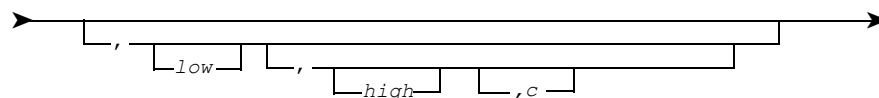


where:

variable is the name of an array.

string is a string.

options is:



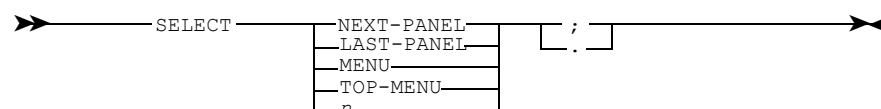
where:

low and *high* are integers.

c is m or M or any other character.

SELECT

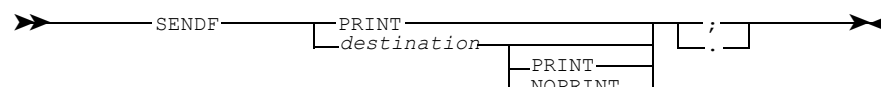
To display an InfoBank panel chosen from the panel you are viewing.



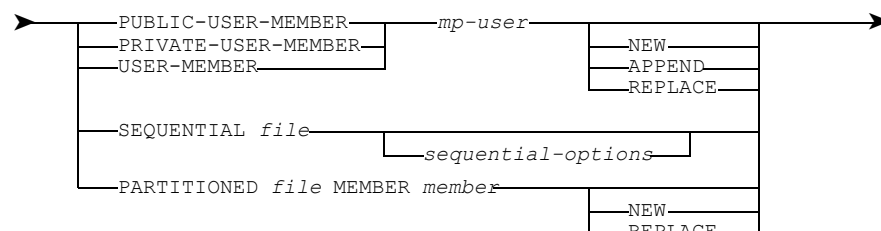
where *n* is an alphanumeric character.

SENDF Executive Command

To direct output from WRITEF directives to a specified destination.



where *destination* is:

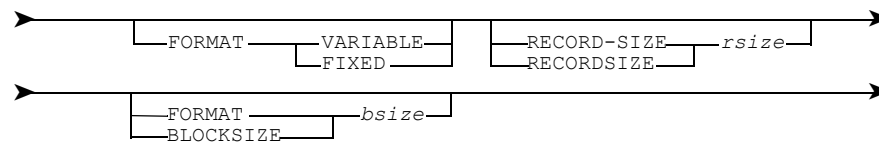


where:

mp-user is the name of an MP-AID USER-MEMBER.

file is the name of a sequential or partioned dataset. It is the logical file name (ddname or dtfname) used in job control statements to define the external dataset name (physical file name) of the file.

sequential-options are:



where:

rsize is the record length.

bsize is the block size.

member is the name of the member of the partitioned dataset.

SET Directive

To assign the result of an evaluated expression to the specified variable.

➤➤ `SET variable-name expression` ➤➤

where:

variable-name is the name of a variable.

expression is an expression subject to Full Evaluation.

SET ALERT

To set the audible alert on or off.

➤➤ `SET ALERT [ON | OFF] ;` ➤➤

SET AUTOSKIP

To set the Autoskip feature on or off.

➤➤ `SET AUTOSKIP [ON | OFF] ;` ➤➤

SET BLANK-LINE-DISPLAY

To suppress or display blank lines in output to the screen.

➤➤ `SET [BLANK-LINE-DISPLAY | SPACE-LINE-DISPLAY] [ON | OFF] ;` ➤➤

SET BROWSE-EDIT and SET OUTPUT-EDIT

To specify the buffer upon which certain editing commands, when issued from an Executive Routine, are actioned.

```

>>-----SET-----☐OUTPUT-EDIT☐-----☐ON☐-----☐OFF☐-----☐;☐----->>
               ☐BROWSE-EDIT
  
```

SET BROWSE-RETENTION

See ["SET LOOKASIDE-RETENTION and SET BROWSE-RETENTION" on page 162.](#)

SET CHECK-CHARACTER

To specify the check-character.

```

>>-----SET CHECK-CHARACTER-----☐x☐-----☐OFF☐-----☐;☐----->>
  
```

where *x* is any delimited or undelimited character except space.

SET COMMAND-LINE

To control the position of the Command Area on the screen.

```

>>-----SET COMMAND-LINE-----☐TOP☐-----☐BOTTOM☐-----☐;☐----->>
  
```

SET COMMAND-RETENTION

See ["SET EXECUTIVE-RETENTION and SET COMMAND-RETENTION" on page 160.](#)

SET COUNTER

To define the increment for the Line Counter.

```

>>-----SET COUNTER n-----☐;☐----->>
  
```

where *n* is the increment at which the Line Counter is to change.

SET CURSOR-HOLD

To specify whether the cursor is to be retained in its current position on the screen when the ENTER key is used in Edit Mode or Update Mode.

```

>>-----SET CURSOR-HOLD-----☐ON☐-----☐OFF☐-----☐;☐----->>
  
```

SET DIAGNOSTIC-DUMP

To provide a mechanism to control the output generated when a problem occurs. This command should only be used under ASG supervision.

where *option* is BUFFERPOOLS, COMMANDS, LOADED, MODULES, SYSTEM, TDS, or TERMINAL.



SET DIRECTLY-ASSUMPTION

To specify whether subsequent WHAT, WHICH, or DOES commands are to report only direct references to selected members, or both direct and indirect references.

SET DIRECTLY-ASSUMPTION ON OFF ;

SET DOCUWARE

To enable you to examine Infobank panels relating to software that is not included in your Manager Products installation, subject to the terms of a Non-Disclosure Agreement.


 DOCUWARE *selectable-unit-code* ☐ ON ☐ OFF ; 

where *selectable-unit-code* is the code for a selectable unit or a Product Nucleus.

SET ECHO

To set the Command Echo capability on or off.

SET ENQUEUE-TRACE

To provide a report of enqueueing and dequeuing activity during the execution of subsequent commands. The command should only be used under ASG supervision.

SET ENQUEUE-TRACE ON ;

SET ENTER-KEY

To set the ENTER-KEY so that whatever is subsequently keyed into the Command Area is passed as a parameter to an Executive Routing (or set of Executive Routines) each time ENTER is pressed.

➤ SET ENTER-KEY *executive* * ☐ ON ☐ OFF ☐ ; ☐ . ☐ ➤

where *executive* is the name of the Executive Routine to be executed and the asterisk (*) represents the point of substitution for the data entered from the Command Area.

SET ESCAPE-CHARACTER

To specify the escape character.

➤ SET ESCAPE-CHARACTER *x* ☐ ; ☐ . ☐ ➤

where *x* is a delimited or undelimited character.

SET EXCP-MONITOR

To set a display of the number of input/output operations on the repository and MP-AID to a particular integer value or to turn off the display.

➤ SET EXCP-MONITOR ☐ *n* ☐ OFF ☐ ; ☐ . ☐ ➤

where *n* is an integer.

SET EXEC-WRITE

To permit or suppress output to the terminal while an Executive Routine is being executed.

➤ SET EXEC-WRITE ☐ ON ☐ OFF ☐ ; ☐ . ☐ ➤

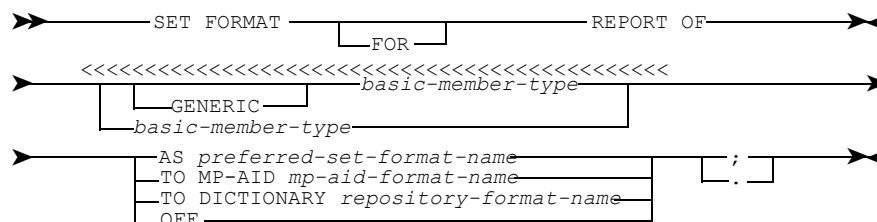
SET EXECUTIVE-RETENTION and SET COMMAND-RETENTION

To specify whether designated Executive Routines are to be retained in virtual storage after execution.

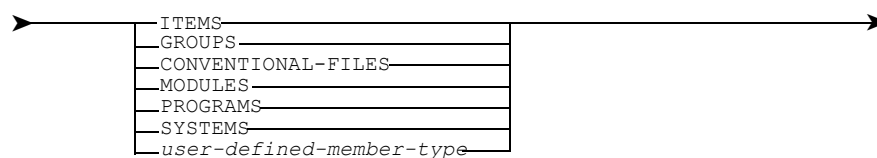
➤ SET ☐ EXECUTIVE-RETENTION ☐ ON ☐ OFF ☐ ; ☐ . ☐ ➤
☐ EXEC-RETENTION ☐ ; ☐ . ☐ ➤
☐ COMMAND-RETENTION ☐ ; ☐ . ☐ ➤
☐ CPU TIME ☐ ; ☐ . ☐ ➤

SET FORMAT

To set the format of reports of members to a preferred set format or to user-defined format, or to set it back to the standard Manager Products format.



basic-member-type is:



where:

user-defined-member-type is based on the above basic member-types.

preferred-set-format-name is:

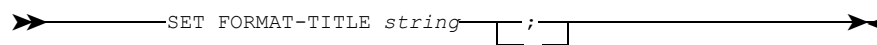


mp-aid-format-name is the name of a FORMAT member on the MP-AID.

repository-format-name is the name of a FORMAT member in the Manager Products repository.

SET FORMAT-TITLE

To specify a printable character string to be included in User Formatted Output. Note that any characters are taken as part of the string, including Manager Products terminators.



where *string* is a string of printable characters up to a maximum of 80 characters long.

SET INFOBANK-SELECT

To control the display of the command keyword SELECT in the command area of an InfoBank panel.

➤ SET INFOBANK-SELECT ☐ ON ☐ OFF ; ☐ .

SET LA-RETENTION

See ["SET LOOKASIDE-RETENTION and SET BROWSE-RETENTION" on page 162.](#)

SET LINE-AREA and SET-PREFIX-AREA

To control the position of the Line Command Area on the screen.

➤ SET ☐ LINE-AREA ☐ PREFIX-AREA ☐ LEFT ☐ RIGHT ☐ OFF ; ☐ .

SET LINE-END-CHARACTER

To set the Line End Character to a user-chosen character.

➤ SET LINE-END-CHARACTER ☐ x ☐ OFF ; ☐ .

where *x* is any single character with the exception of the Manager Products terminators.

SET LINEAR-RETENTION

To set the Linear Retention capability on or off.

➤ SET LINEAR-RETENTION ☐ ON ☐ OFF ; ☐ .

SET LIST-SHIFT

To specify whether the display of member names in output from a LIST command is to be shifted left.

➤ SET LIST-SHIFT ☐ ON ☐ OFF ; ☐ .

SET LOOKASIDE-RETENTION and SET BROWSE-RETENTION

To specify whether Lookaside Buffers are to be retained up to the maximum buffer count, despite subsequent EDIT or UPDATE commands.

➤ SET ☐ LOOKASIDE-RETENTION ☐ BROWSE-RETENTION ☐ LA-RETENTION ☐ ON ☐ OFF ; ☐ .

SET NUMBERS

To set the Numbers feature for source lines of repository members on or off.

```

>>-----SET NUMBERS-----[ON] [OFF] [;]----->>

```

SET OUTPUT-EDIT

See ["SET BROWSE-EDIT and SET OUTPUT-EDIT" on page 158](#).

SET PAGE-HEADINGS

To indicate whether the standard Manager Products header or a header with specified user-defined text is to be used in page headings in output generated using the SWITCH OUTPUT command or from Manager Products batch runs for subsequent output to a hard copy device.

```

>>-----SET PAGE-HEADING-----[ON]----->>
                                [OFF]----->>
                                [page-heading]-----[RESET]----->>
                                                    [NORESET]----->>
                                                    [NO-RESET]----->>
                                                    [NOPAGE]----->>
                                                    [NO-PAGE]----->>
<-----[;]-----<

```

where *page-heading* represents a string of printable characters up to a maximum of 100 characters long.

SET PAGE-LENGTH

To specify the length of pages generated using the SWITCH OUTPUT command or from Manager Products batch runs, for subsequent output to a hard copy device.

```

>>-----SET PAGE-LENGTH n-----[;]----->>

```

where *n* may be null, 0, or an integer in the range 7 to 255.

SET PERFORM-CHARACTER

To set the Perform Character for PERFORM commands and COMMAND-STREAM members.

```

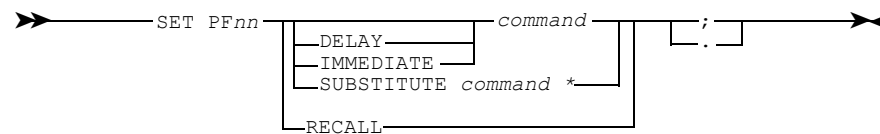
>>-----SET PERFORM-CHARACTER x-----[;]----->>

```

where *x* is a delimited or undelimited character.

SET PFnn

To set a PF key to command.



where:

nn is the number of the key to be set.

command is any Manager Products command available to the user, or the name of an Executive Routine.

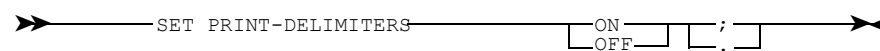
* ensures that values for PF keys entered in the Command Area are implemented when the SUBSTITUTE keyword is used.

SET PREFIX-AREA

See ["SET LINE-AREA and SET-PREFIX-AREA" on page 162](#).

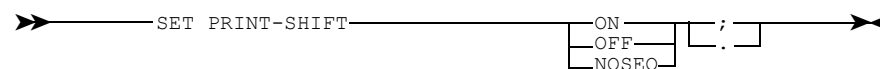
SET PRINT-DELIMITERS

To set the PRINT OF . . . /END of PRINT delimiters for prints of repository and MP-AID members on or off.



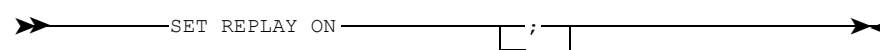
SET PRINT-SHIFT

To set the output from the PRINT and BULK PRINT commands so that leading spaces and/or source line numbers can be displayed or suppressed.



SET REPLAY ON

To resume an interrupted PLAY mode session.



SET REVISION-BARS

To display or suppress the indication of changes to the test of an InfoBank panel by means of a Revision Bar or Deleted Line Indicator in the left-hand column.

➤ — SET REVISION-BARS — ☐ ON ☐ OFF ☐ ; ☐ — ➤

SET SCALE

To set the Scale feature of the General Purpose Screen on or off.

➤ — SET SCALE — ☐ ON ☐ OFF ☐ ; ☐ — ➤

SET SCREEN-REFRESH

To set the Screen Refresh capability to scroll continuously through output as it is being generated, or to set it off.

➤ — SET SCREEN-REFRESH — ☐ *nn* ☐ OFF ☐ ; ☐ — ➤

where *nn* is an number of lines.

SET SCROLL-LIMIT

To set a value for the Scroll Limit capability (retention of lines in Command Mode Buffer), or to set it off.

➤ — SET SCROLL-LIMIT — ☐ *n* ☐ OFF ☐ ; ☐ — ➤

where *n* is any number, subject to the limits on your allocation of virtual storage.

SET SOS-DUMP

To initiate a dump. This command should only be used under ASG direction.

➤ — SET SOS-DUMP — ☐ ON ☐ OFF ☐ ; ☐ — ➤

SET SPACE-LINE-DISPLAY

See ["SET BLANK-LINE-DISPLAY" on page 157](#).

SET SUBSTITUTION CHARACTER

To specify the substitution character to be used with the substitutable PF key facility.

```

>>-----SET SUBSTITUTION-CHARACTER character [ ] ; [ ]----->>

```

where *character* is the required substitution character.

SET TAB-CHARACTER

To specify the tab character.

```

>>-----SET TAB-CHARACTER x [ ] ; [ ]----->>

```

SET TABS

To specify tab positions.

```

>>-----SET TABS-----<<<<<<<<
[ ] nn [ ] ; [ ]----->>

```

where *nn* is the column position defining the start of a tab.

SET TRACE

To display information as executive routines run, to help you debug those routines; and to provide general Manager Products debug information for use in problem resolution.

```

>>-----SET TRACE----->>
>-----OFF----->>
|-----|
| procl-trace |
| general-trace |
| DNAME ddname |
| [ ] * |
|-----|
>-----[ ] ; [ ]----->>

```

where *procl-trace* is:

```

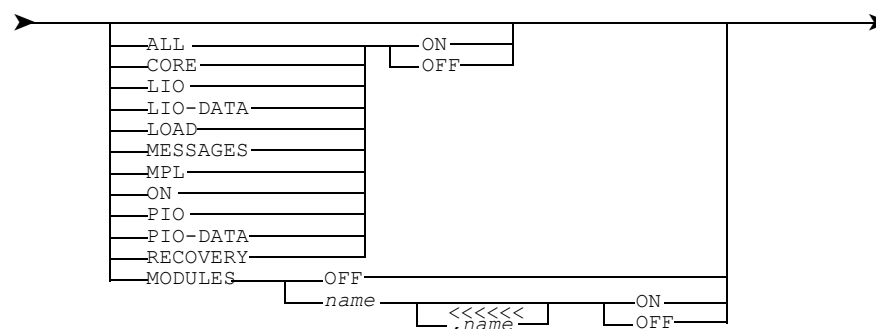
>----->>
| ASSIGNS | ON |
| CPUTIME | OFF |
| INSTRUCTIONS |
| LOGIC |
| MPR |
| COMMANDS |
| RESULTS |
| PROCEDURES | OFF |
| VARIABLES | name |
|-----|
|-----<<<<<<
| , name | ON |
|-----|
|-----OFF----->>

```

where:

name is the name of a procedure (maximum 10 characters) or a variable (maximum 50 characters).

general-trace is:



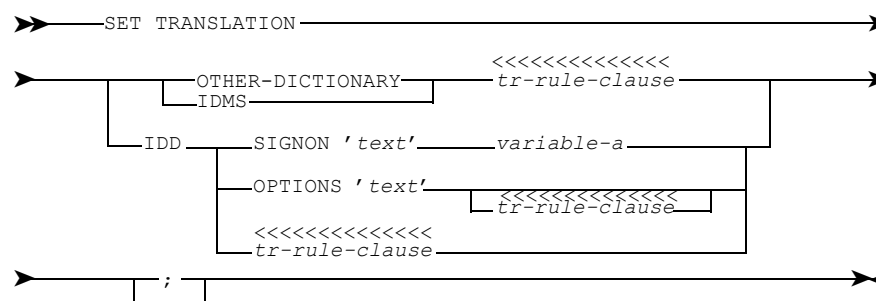
where:

name is the name of a module (maximum 8 characters).

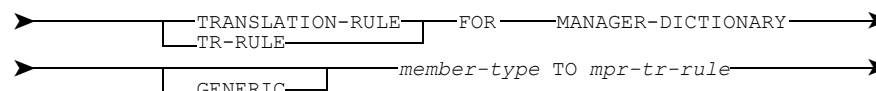
ddname is a DDNAME having a maximum of 8 characters.

SET TRANSLATION

To specify the default Translation Rules which are to apply during the transfer of repository members to other repositories or directories.



where *tr-rule-clause* is:



where *member-name* is the name of a repository member.

SHOW ALIAS-TYPES

To display information about general aliases and specific alias-type keywords.

```

>>-----SHOW ALIAS-TYPES----->>
                |   FOR INPUT   |
                |   FOR MSP    |
  
```

SHOW MEMBER-TYPE

To display the variables for and structure of specified member types.

```

>>-----SHOW MEMBER-TYPE----->>
                |   TABLE x   |
                |   VARIABLE   |
                |   STRUCTURE  |
                |   NAMES      |
>-----FOR-----MEMBER-TYPE member-type----->
|   BASED   |   COMMON-CLAUSES   |   ;   |
  
```

SHOW PRODUCE-OPTIONS

To display the currently active settings for each control option of the PRODUCE command.

```

>>-----SHOW PRODUCE-OPTIONS FOR----->>
>-----TOTAL----->
| ADABAS |
| MARKIV |
| MARK-IV |
| SYSTEM2000 |
| S2K    |
|-----|
| COBOL  |
| PL/I   |
| PL/1   |
| PLI    |
| PL1    |
| BAL    |
| ASSEMBLER |
| ASSEMBLY |
| ALC    |
| RECORD-LAYOUTS |
|-----|
| IMS    |
| DL/I   |
| DL/1   |
| DLI    |
| DL1    |
|-----|
|-----SEGMENT-OPTIONS-----|
| IOGEN  |
|-----|
|-----DATABASE-DESCRIPTIONS-----|
| DBDGEN |
| PROGRAM-SPECIFICATIONS |
| PSBGEN |
  
```

SHOW UDR

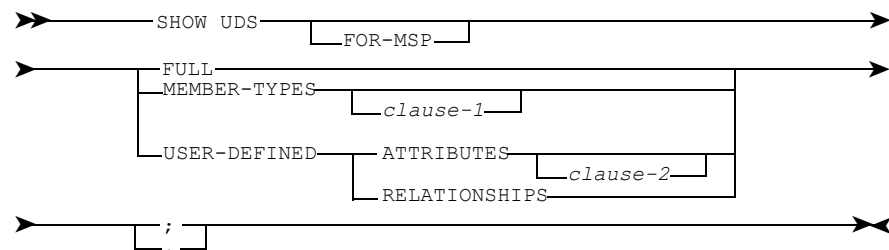
To show the default or user-defined UDR clauses available in your repository.

```

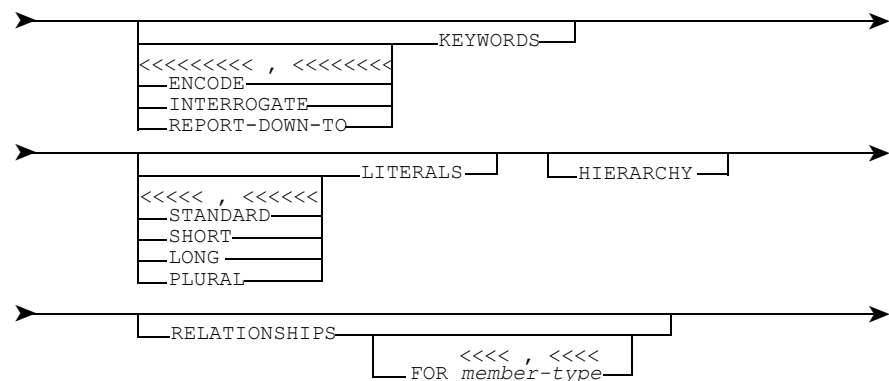
>>-----SHOW UDR----->>
                |   ;   |
  
```

SHOW UDS

To display all or part of the repository schema.



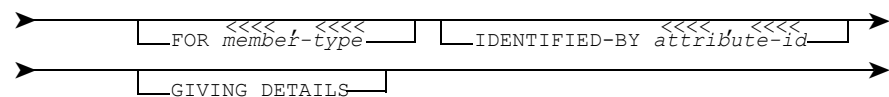
where *clause-1* is:



where:

member-type is the interrogate keyword for any member type, or any collective member-type, available in your repository.

clause-2 is:



where:

member-type is as defined above.

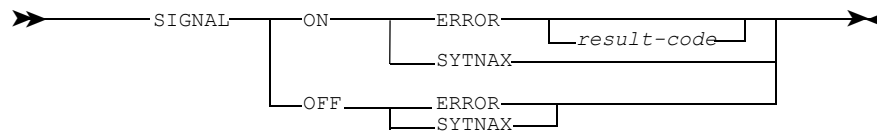
attribute-id is a user-defined attribute type.

Note:

The FOR MSP option is provided for Product Support purposes and should only be used under instruction from ASG.

SIGNAL Directive

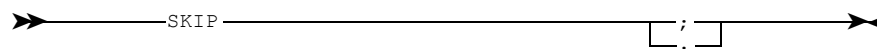
To allow you to divert control within an Executive Routine when an error occurs.



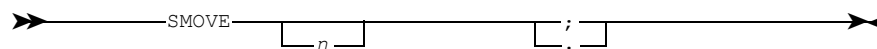
where *result-code* is an expression that evaluates to any integer between 0 and 255 inclusive.

SKIP

To cause the next line of printed output, if directed to a line printer, to be printed at the head of a new page.

**SMOVE**

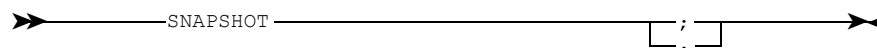
To move lines from the current buffer to the Scratchpad without overwriting the contents of the Scratchpad. The move starts from the current line.



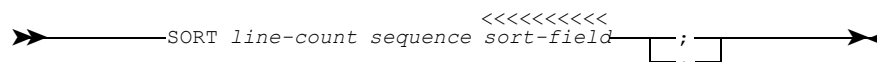
where *n* is an integer.

SNAPSHOT

To produce a summary of the data in the Workbench Design Area.

**SORT**

To sort lines of text in any buffer, and put the sorted output into an Edit Buffer.



where:

line-count is the number of lines to be sorted (* denotes all).

sequence is A for Ascending or D for Descending sort order.

sort-field comprises the start column and end column of the sort field, up to 50 of which may be specified.

SOURCE-FILE

See ["SFILE and SOURCE-FILE" on page 168](#).

SPACE

To cause the next line of printed output, if directed to a line printer, to be printed after a stated number of blank lines.

```

>>-----SPACE-----[integer]-----[.];-----><

```

where *integer* is an unsigned integer in the range 1 to 99.

SQL ACQUIRE

To generate an SQL ACQUIRE DBSPACE statement.

```

>>-----SQL ACQUIRE dbspace-----[destination]-----[.];-----><

```

where:

dbspace is the name of a SQL DBSPACE member.

destination is:

```

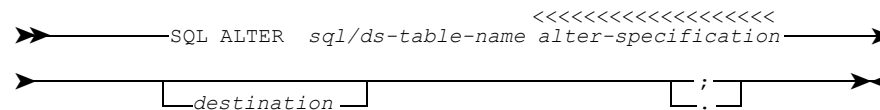
>-----ONTO-----[PRIVATE]-----[PUBLIC]-----mpaid-member-name-----[NEW]-----[APPEND]-----[REPLACE]----->
>-----[PRINT]-----[NOPRINT]----->

```

where *mpaid-member-name* is the name of a USER-MEMBER.

SQL ALTER

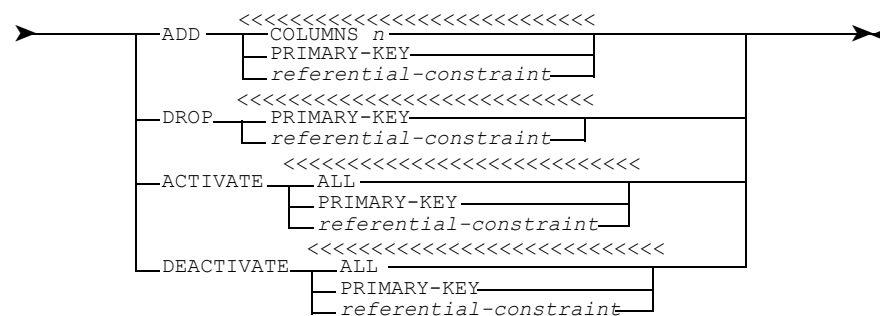
To generate one or more SQL ALTER TABLE statements.



where:

sql/ds-table-name is the name of a SQL-TABLE member.

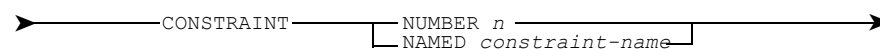
alter-specification is:



where:

n is an integer.

referential-constraint is:

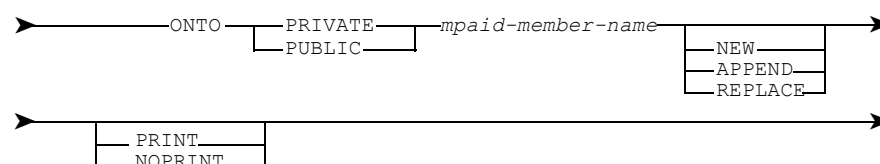


where:

n is as defined above.

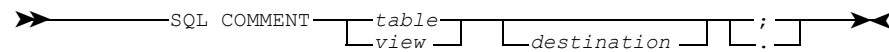
constraint-name is a name specified in the NAMED clause of an SQL-TABLE member.

destination is:



SQL COMMENT

To generate SQL COMMENT ON statements.

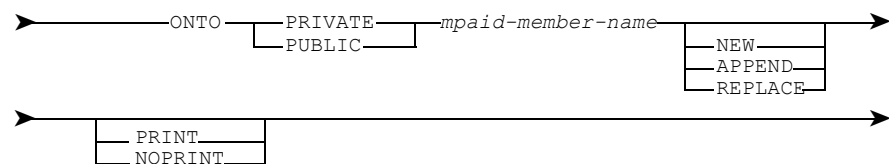


where:

table is the name of a SQL-TABLE member.

view is the name of a SQL-VIEW member.

destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

SQL CREATE

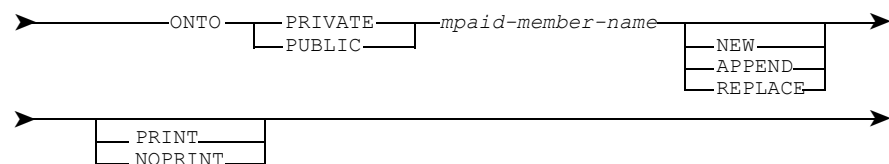
To generate an SQL CREATE statement.



where:

member is the name of a SQL-INDEX, SQL-TABLE, or SQL-VIEW member.

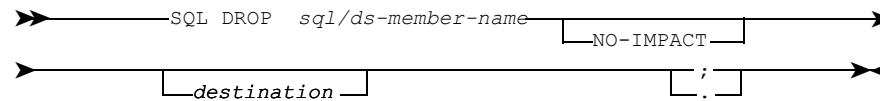
destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

SQL DROP

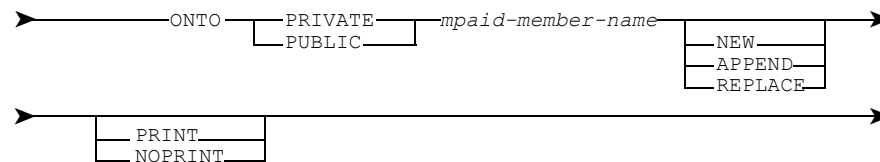
To generate an SQL DROP statement. An impact analysis report is generated displaying the impact the SQL DROP statement will have in your SQL/DS environment.



where:

sql/ds-member-name is the name of a SQL-DBSPACE, SQL-INDEX, SQL-TABLE, or SQL-VIEW member.

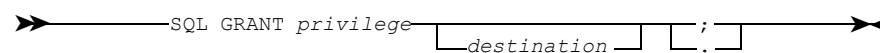
destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

SQL GRANT

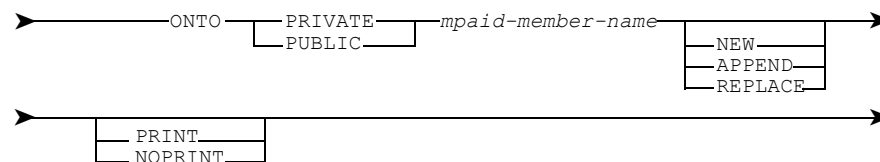
To generate SQL GRANT statements.



where:

privilege is the name of a SQL-PRIVILEGE member.

destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

SQL LABEL

To generate SQL LABEL ON statements.

SQL LABEL ☐ *table* ☐ *view* ☐ *destination* ☐ ; ☐

where:

table is the name of a SQL-TABLE member.

view is the name of a SQL-VIEW member.

destination is:

ONTO ☐ PRIVATE ☐ *mpaid-member-name* ☐ NEW ☐
☐ PUBLIC ☐ ☐ APPEND ☐
☐ REPLACE ☐
☐ PRINT ☐
☐ NOPRINT ☐

where *mpaid-member-name* is the name of a USER-MEMBER.

SQL LIST CYCLES

To identify the cycles found in the SQL/DS design present in the Workbench Design Area (WDBA) and to list the tables which appear within each cycle.

SQL LIST CYCLES ☐ ALPHABETICALLY ☐ ; ☐

SQL LIST TABLES

To produce a list of all or some of the tables appearing in the SQL/DS design generated in the Workbench Design Area (WDBA).

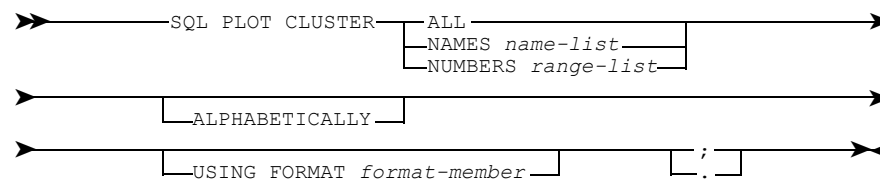
SQL LIST TABLE ☐ ALPHABETICALLY ☐
☐ *table-type* ☐ ; ☐

where *table-type* is:

☐ ROOTS ☐ PARENTS ☐ LEAFS ☐
☐ LEAVES ☐
☐ DEPENDENTS ☐ INDEPENDENTS ☐

SQL PLOT CLUSTER

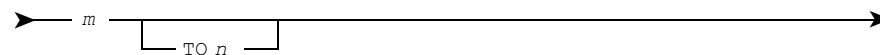
To produce a SQL Cluster Plot of all or some of the tables in the SQL design.



where:

name-list is a list of validity named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:

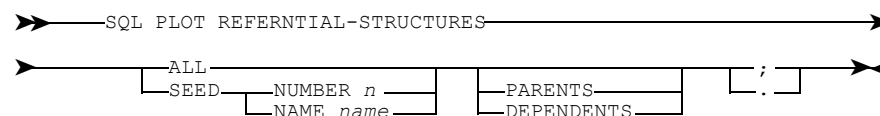


where *m* and *n* are valid WBDA table numbers and *n*, if it appears, is greater than *m*.

format-member is the name of a previously defined, valid format member.

SQL PLOT REFERENTIAL-STRUCTURES

To produce a SQL Referential Structures Plot of all or a single one of the referential structures in the SQL/DS design.



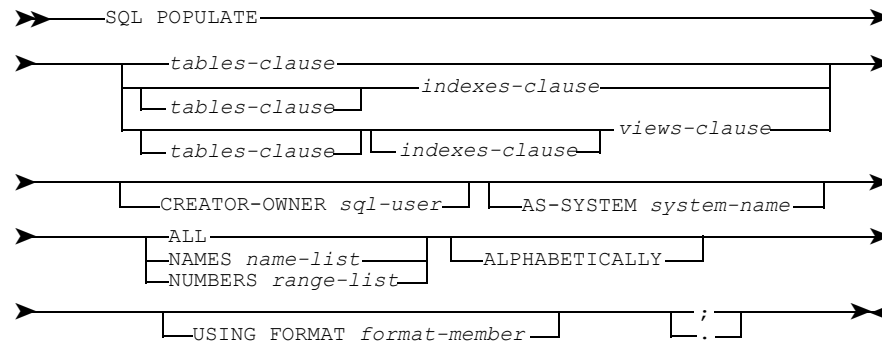
where:

n is a table number in the Workbench Design Area.

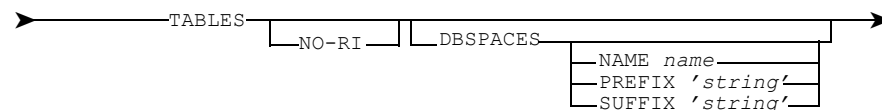
name is a table name.

SQL POPULATE

To populate the repository with SQL-TABLE, SQL-INDEX, and SQL-VIEW members, generated from the SQL/DS design, and optionally to produce a report of the generated members.



where *tables-clause* is:

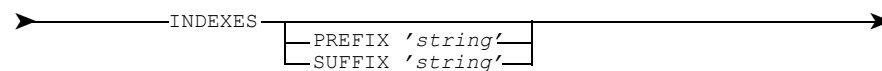


where:

name is an alphanumeric string of up to 32 characters which should conform to the rules for a valid Manager Products repository member name.

string is an alphanumeric string of up to 31 characters which should conform to the rules for a valid Manager Products repository member name.

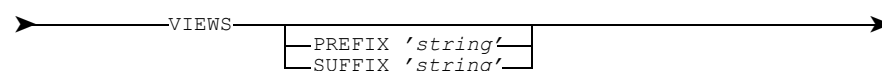
indexes-clause is:



where:

string is defined as above.

views-clause is:



where:

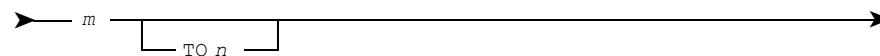
string is defined as above.

sql-user is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

system-name is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

name-list is a list of validly named tables in the WDDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:

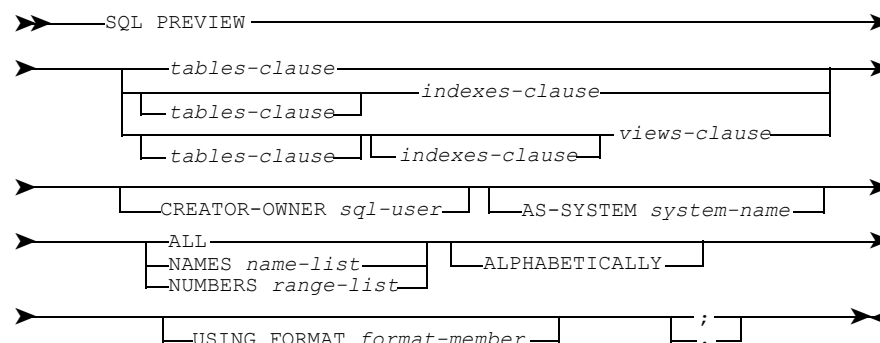


where *m* and *n* are valid WBDA table numbers and *n*, if it appears, is greater than *m*.

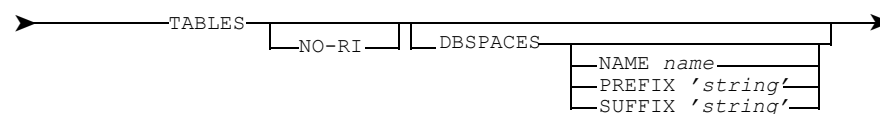
format-member is the name of a previously defined, valid format member.

SQL PREVIEW

To generate and report repository members from SQL tables, indexes, and views in the SQL/DS design.



where *tables-clause* is:

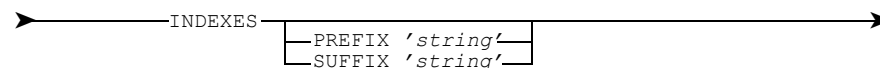


where:

name is an alphanumeric string of up to 32 characters which should conform to the rules for a valid Manager Products repository member name.

string is an alphanumeric string of up to 31 characters which should conform to the rules for a valid Manager Products repository member name.

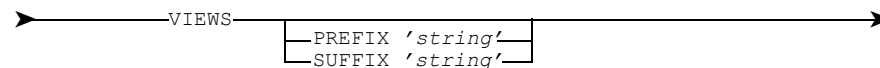
indexes-clause is:



where:

string is defined as above.

view-clause is:



where:

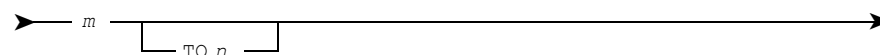
string is defined as above.

sql-user is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

system-name is an alphanumeric string of up to 32 characters, which should conform to the rules for a valid Manager Products repository member name.

name-list is a list of validly named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:

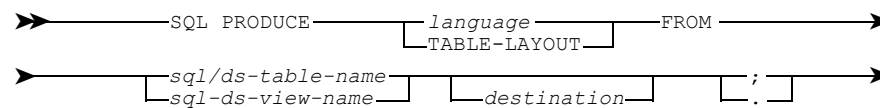


where *m* and *n* are valid WBDA table numbers and *n*, if it appears, is greater than *m*.

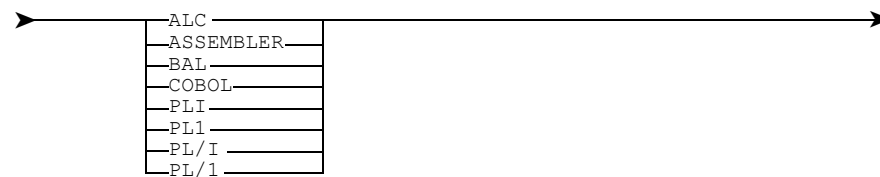
format-member is the name of a previously defined, valid format member.

SQL PRODUCE

To generate a host language data structure or a table layout.



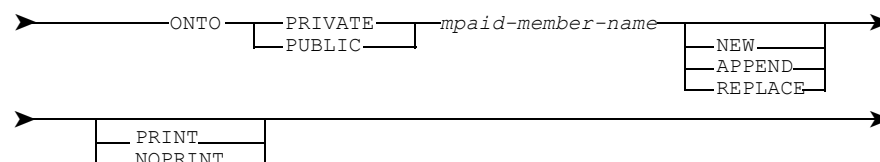
where *language* is:



sql/ds-table-name is the name of a SQL-TABLE member.

sql/ds-view-name is the name of a SQL-VIEW member.

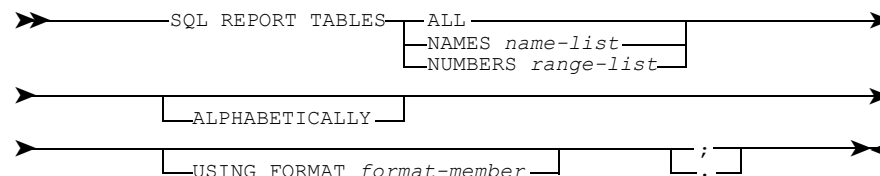
destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

SQL REPORT

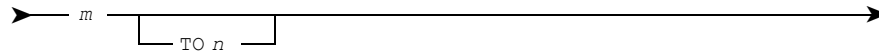
To produce an SQL Table Report of all or some of the tables in the SQL/DS design.



where:

name-list is a list of validly named tables in the WBDA. If there are two or more names in the list they must be separated by commas.

range-list is a list of one or more numeric ranges, separated by commas, each of the form:



where *m* and *n* are valid WBDA table numbers and *n*, if it appears, is greater than *m*.

format-member is the name of a previously defined, valid format member.

SQL REVOKE

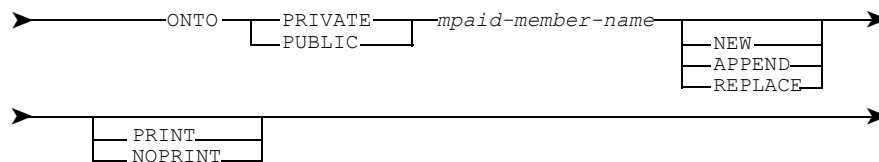
To generate SQL REVOKE statements.



where:

privilege is the name of a SQL-PRIVILEGE member.

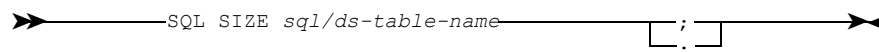
destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

SQL SIZE

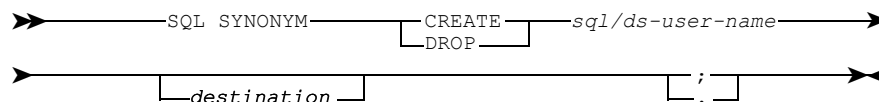
To calculate the total size of a table and the maximum size of each row contained in that table.



where *sql/ds-table-name* is the name of a SQL-TABLE member.

SQL SYNONYM

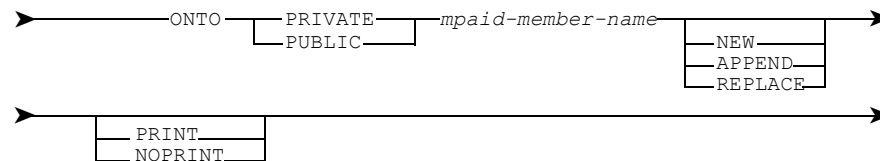
To generate SQL CREATE SYNONYM or DROP SYNONYM statements.



where:

sql/ds-user-name is the name of a SQL-USER member.

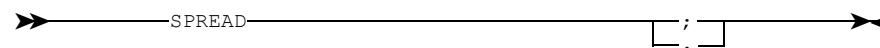
destination is:



where *mpaid-member-name* is the name of a USER-MEMBER.

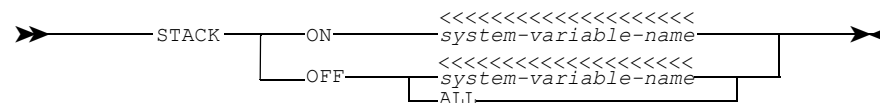
SREAD Procl Executive Command

To enable procedures to read input from the Manager Products primary input device.



STACK Directive

To switch the stacking of specified System Variables ON or OFF.



where *system-variable-name* is the name of any valid System variable except for &COLO, &CURL, &ENAM, &LINC, &LINO, and &PNUM.

STATUS

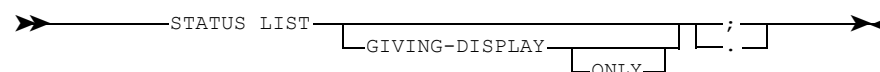
To find the current status and to move from the current status to another.



where *status-name* is the name of a status.

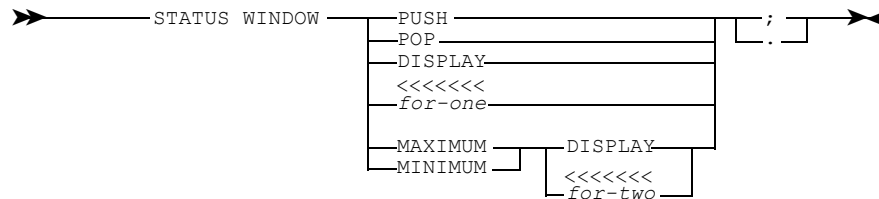
STATUS LIST

To list the statuses available to a user in the repository.

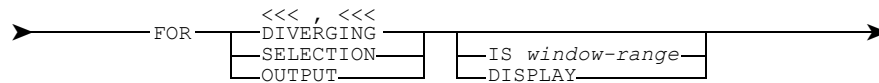


STATUS WINDOW

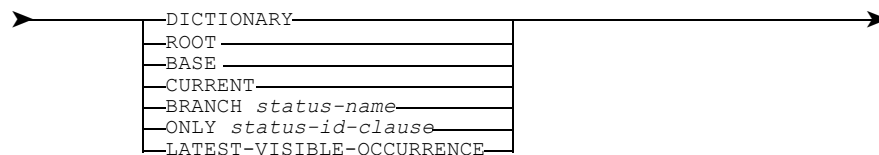
To adjust a user's Status Window within limits (maximum and minimum) established by the Systems Administrator and to display the current Status Window.



where *for-one* is:



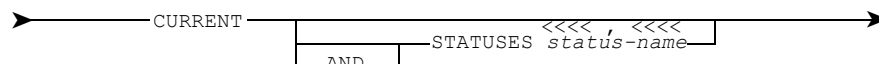
where *window-range* is:



where:

status-name is the name of a status.

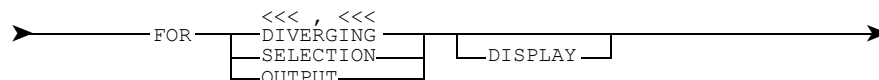
status-id-clause is:



where:

status-name is as defined above.

for-two is:

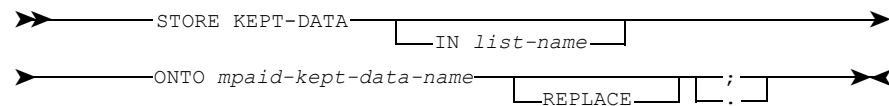


Note:

The window-range LATEST-VISIBLE-OCCURRENCE is only valid for the SELECTION and OUTPUT window-types.

STORE KEPT-DATA

To store named and unnamed KEPT-DATA lists on the MP-AID.



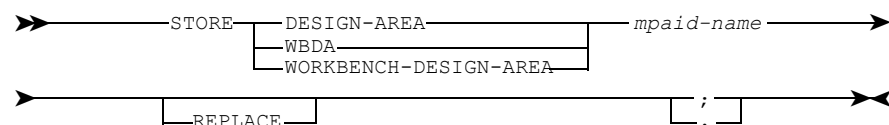
where:

list-name is the name of a KEPT-DATA list.

mpaid-kept-data-name is the name of the MP-AID KEPT-DATA member in which the KEPT-DATA is to be stored.

STORE WORKBENCH-DESIGN-AREA

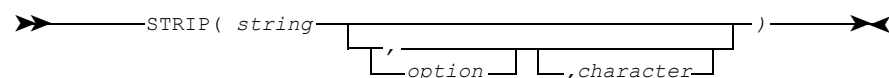
To store in the MP-AID the current content of the Workbench Design Area.



where *mpaid-name* is the name of a Workbench member on the MP-AID.

STRIP Function

To remove leading, trailing, or both leading and trailing characters from a string.



where:

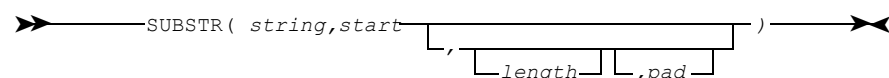
string is a string.

option is one of L, T, or B.

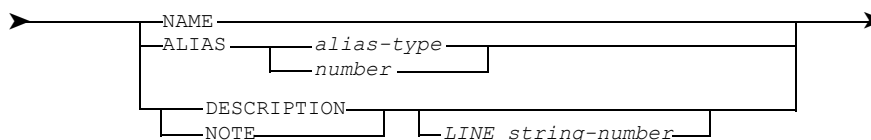
character is a character.

SUBSTR Function

To return a substring of a given string.



where *identification-element* is:



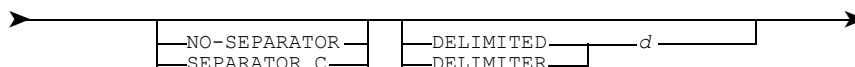
where:

alias-type is a keyword from the specific alias-type keyword list of the repository.

number is an unsigned integer in the range 1 to 16.

string-number is an unsigned integer in the range 1 to 32767.

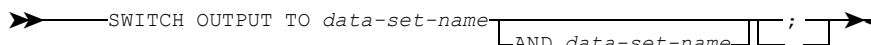
variable-b is:



where *c* and *d* are single printable characters and must not be space characters.

SWITCH OUTPUT

To switch output to an additional or alternative dataset.



where *data-set-name* is the logical file name (ddname or dname) used in OS job control statements to indicate the name of the external dataset (physical file name) to which output is to be written (in CMS, TSO, Siemens Timesharing Interface, and OS batch environments).

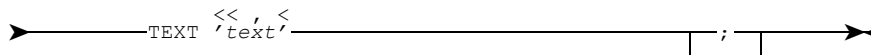
In CICS environments ddname is any 4-character destination identification.

TEXEC

See ["TARNSIENT-EXECUTIVE or TEXEC" on page 189](#).

TEXT

To output user-supplied text in the context of an Executive Routine, a PERFORM command, or a POST or MAIL command.



where *text* is a character string of not more than 255 characters.

TO AUPD

TOAUPD converts a native update (UPD) buffer into an assisted update (AUPD) buffer. This might be helpful in the event of an encode failure, allowing the member definition to be displayed in AUPD format.

```
➤➤——TO AUPD——➤➤
```

TOP

To display the first screen of the current buffer.

```
➤➤——TOP——;——➤➤
      |_____|
```

TRACE Directive

To display selected lines as an Executive Routine runs. Use it to debug Executive Routines.

```
➤➤——TRACE——<<<<OFF>>>>——➤➤
              |____options____|
```

where *options* is one of OFF, ALL, ASSIGNS, COMMANDS, LOGIC, ON, or RESULTS.

TRANSFER Command

To transfer data from the MP-AID to a file external to Manager Products.

```
➤➤——TRANSFER FROM USER-MEMBER mp-user——➤➤
                                     |____LOGON logon-id____|
➤➤——TO FILE filename——PARTITIONED——➤➤
                             |____SEQUENTIAL____|
                                     |____'control-card'____|
                                     |____NO-CONTROL-CARD____|
➤➤——AS member-name——;——➤➤
      |____|
```

where:

mp-user is the name of a USER-MEMBER on the MP-AID which holds the data to be transferred.

logon-id is the LOGON-ID of the user who created the specified USER-MEMBER.

filename is the name of the output dataset. It is the logical filename (ddname or dtfname) used in job control statements to define the external dataset name (physical filename) of the file to which the USER-MEMBER is to be copied.

control-card is a character string of up to 72 characters, being a library system control card image. Trailing spaces are implied. A single question mark (?) character may be used to indicate the point at which the generated library member name is to be inserted in the control card.

member-name is the name of the generated member in the output dataset.

TRANSFER Directive

To cause the current Executive Routine and all higher level Executive Routines to be exited from without further processing and then cause the named Executive Routine to start running.

```

>>-----TRANSFER-----exec-name-----<<<<<<<<<
                                   |parameter|

```

where:

exec-name is the name of Executive Routine.

parameter is a parameter to be passed to the named Executive Routine.

TARNSENT-EXECUTIVE or TEXEC

To execute a Transient Executive Routine.

```

>>-----TRANSIENT-EXECUTIVE-----name-----<<
|TEXEC|                                     |.|

```

where *name* is the name of a Corporate Executive Routine.

TRANSLAT Function

To translate specified characters in *string* into other characters or reorder them.

```

>>-----TRANSLAT( options )-----<<

```

where *options* is:

```

>-----string,-----,inp-----,pad----->
               |out|

```

where:

string is a string.

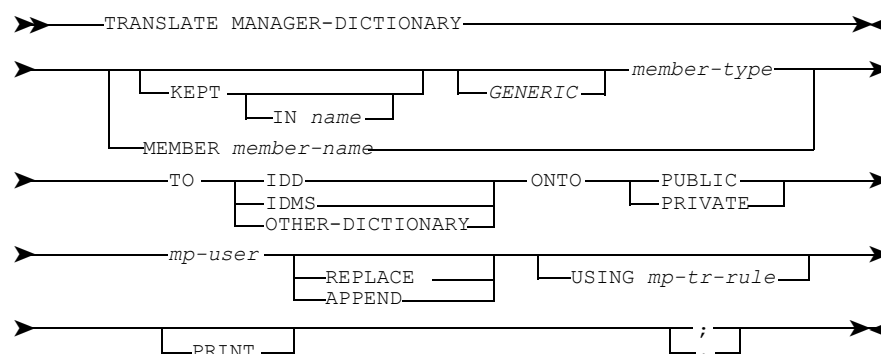
out is the output translation table.

inp is the input translation table.

pad is a character.

TRANSLATE

To translate Manager Products repository member definitions to the format of another repository or directory.



where:

name is the name of the KEPT-DATA list of members selected for translation.

member-type is a member-type interrogate keyword valid on the current repository. This may be any valid basic member-type, or any UDS member-type based on the basic member-types.

member-name is the name of the repository member to be translated.

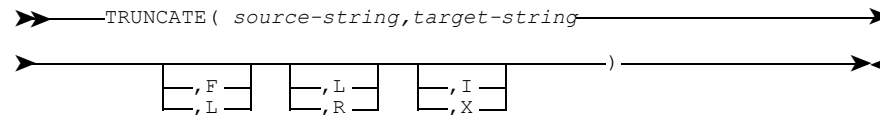
mp-user is the name to be used for a USER-MEMBER on the MP-AID. The name must be no more than ten characters long. A public USER-MEMBER can be accessed by someone with a different logon-id; a private member cannot.

mp-tr-rule is the name of a TR-RULE member on the MP-AID.

If the **MEMBER** clause is specified, the **USING** clause is required.

TRUNCATE

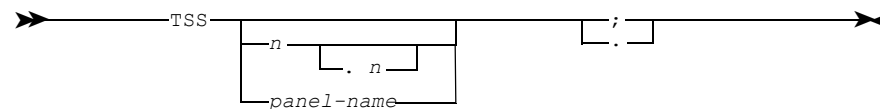
To truncate the source-string at the point given by the target string. Truncation can be either from the left or right of the source string and the target string can be included in or omitted from the truncation.



where *source-string* and *target-string* are strings.

TSS

To start ToolSet SERVICES.



where:

n is an option on a ToolSet SERVICES menu.

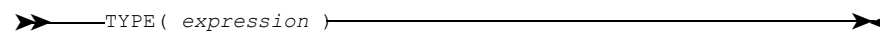
panel-name is the name of a ToolSet SERVICES panel.

Note:

The TSS command can only be entered in the command interface environment.

TYPE Function

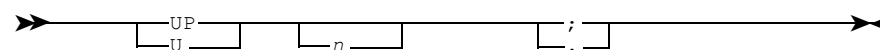
To return the type of an expression.



where *expression* evaluates to a string.

UP

To move the window on the text displayed on the screen up *n* lines.



where *n* is the number of lines to be scrolled backward in a buffer.

UPDATE

To create a new repository member or update an existing one.

```
➤➤ UPDATE [member-name] [ ]; ➤➤
```

where *member-name* is the name of an existing repository member.

UPPER Function

To return a string with all characters translated to uppercase.

```
➤➤ UPPER( string ) [ , I ] [ O ] ➤➤
```

where *string* is a string.

US and USAGE

To list all repository members which refer directly to a member.

Line Command:

```
➤➤ US ➤➤
```

Primary Command:

```
➤➤ USAGE member-name [ ]; ➤➤
```

USA

To list all repository members which refer directly or indirectly to a member.

Line Command:

```
➤➤ USA ➤➤
```

Primary Command:

```
➤➤ USA member-name [ ]; ➤➤
```

USR

To find the members that refer to and are referenced by a specified member via ER relationships.

where:

str is a string of up to 25 characters is a user-defined variable name.

variable-name is the name of any user-defined or system variable, except for the following system variables: &BUFN, &COLO, &ENAM, &LINC, &LINO, and &PNUM.

WAL

To list all repository members which have specified alias.

Line Command:

➤➤ `-WAL` _____ ➤➤

Primary Command:

➤➤ `-WAL alias` _____ `[] ; []` _____ ➤➤

WBDA

To provide a temporary Workbench Design Area mode for the duration of the current command.

➤➤ `-WBDA command` _____ `[] ; []` _____ ➤➤

where *command* is LIST, MERGE, or REPORT.

WF

To list all members with a specified catalog classification.

Line Command:

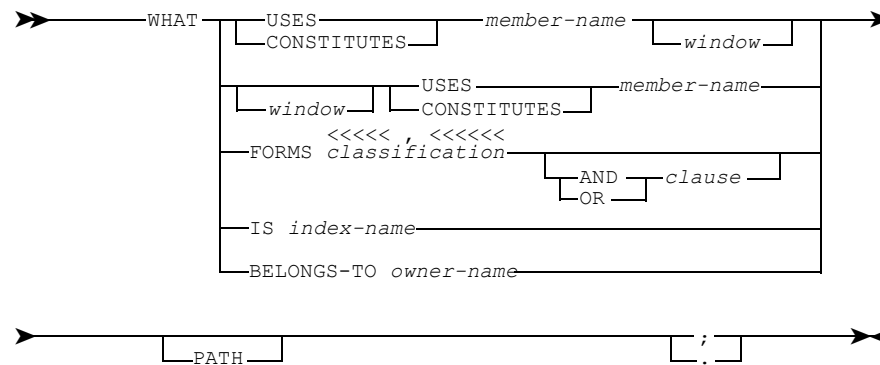
➤➤ `-WF` _____ ➤➤

Primary Command:

➤➤ `-WF catalog-classification` _____ `[] ; []` _____ ➤➤

WHAT

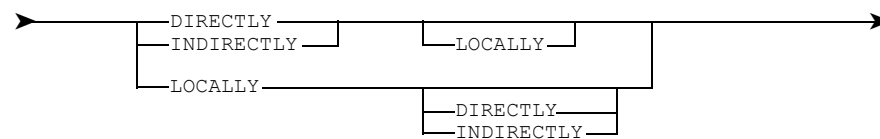
To output index-names that meet specific conditions, or to identify index-names.



where:

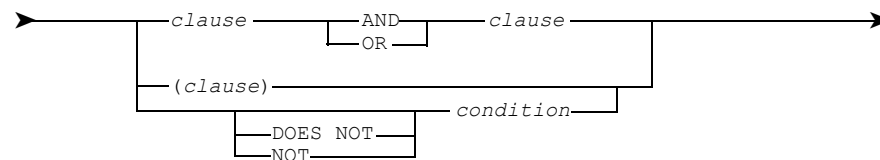
member-name is the name of a repository member.

window is:



classification is a catalog classification.

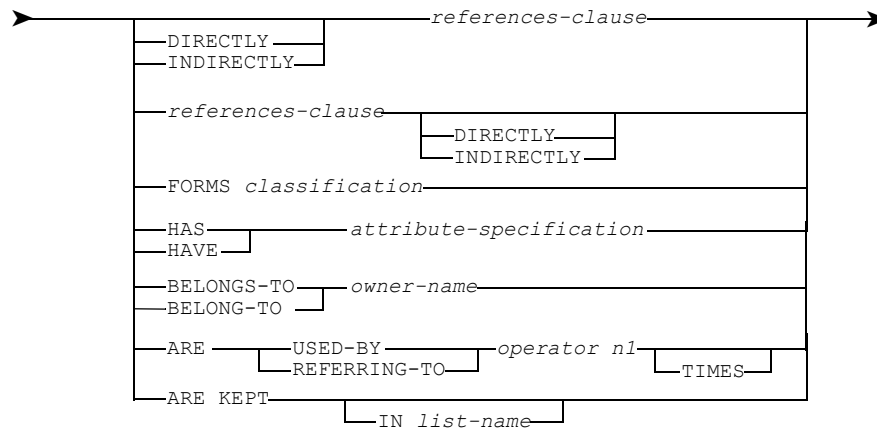
clause is:



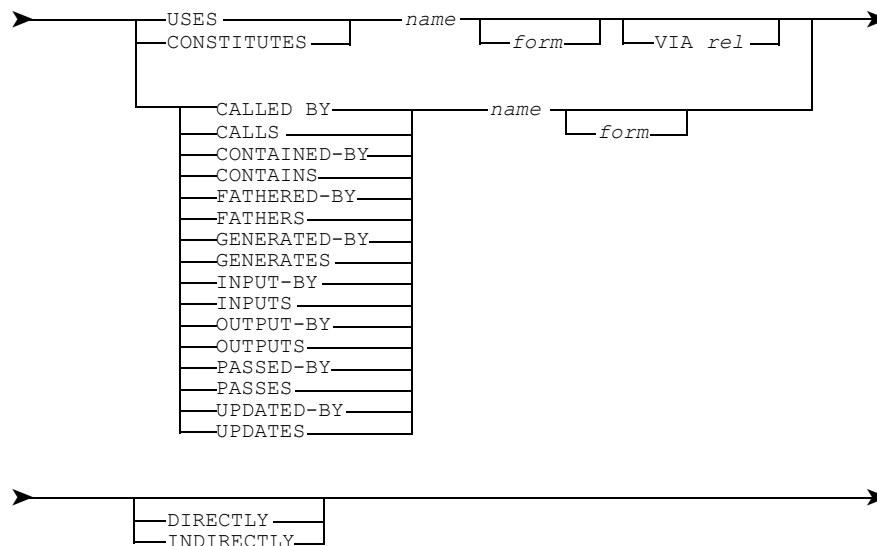
Note:

Clause is recursive.

where *condition* is:



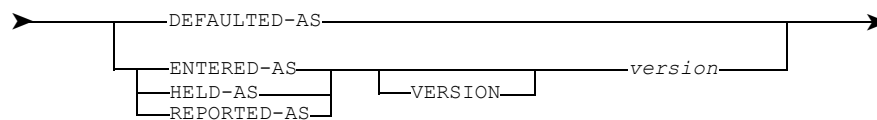
where *references-clause* is:



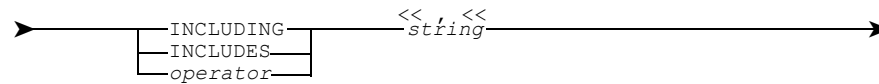
where:

name is the name of a repository member.

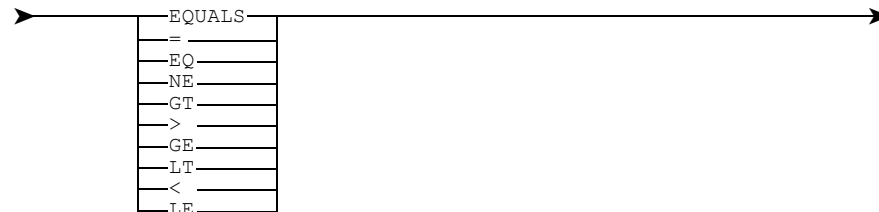
form is:



s-relation is:

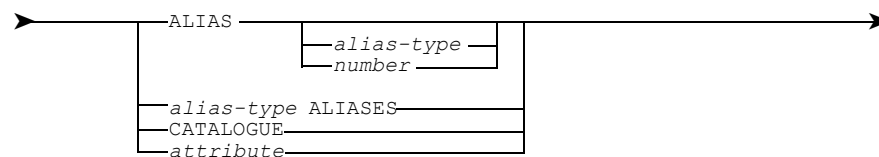


where *operator* is:



string is as defined above.

value-attr is:



where:

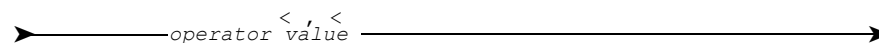
alias-type is one of the specific *alias-type* keywords available in your repository.

number is an integer from one to the maximum number of general aliases allowed in your repository.

Use the `SHOW ALIAS-TYPES` command to find out the specific alias-types and the number of general aliases available in your repository.

attribute is any attribute in a member definition that contains neither text nor a string (see output of SHOW MEMBER-TYPE command).

v-relation is:



where:

operator is as defined above.

value is any variable appropriate to the value-attribute.

owner-name is the name of an owner.

operator is as defined above.

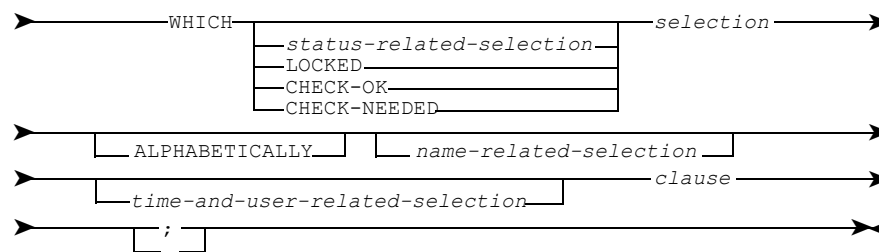
n1 is an unsigned integer in the range 0 to 2,147,483,647.

list-name is the name of a KEPT-DATA list.

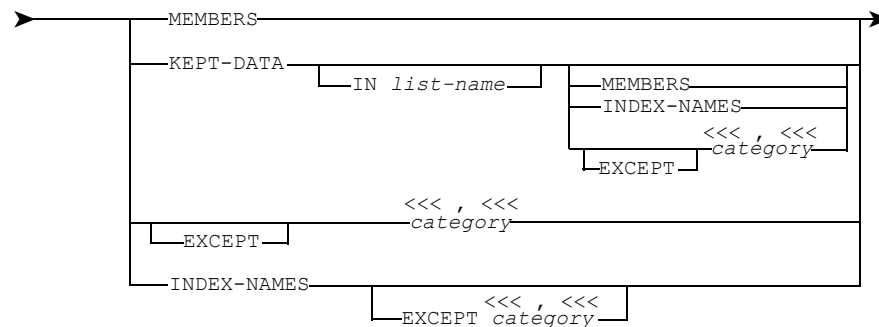
index-name is the name of a repository member, or an alias or a catalog classification, or a user-defined indexed attribute.

WHICH

To select index-names that do or do not meet a condition or set of conditions.



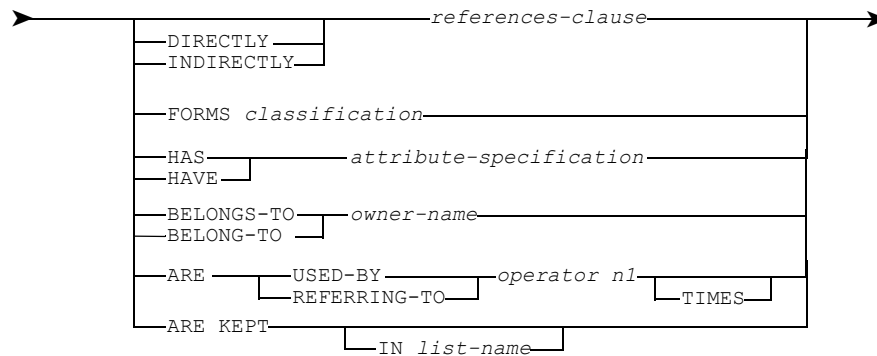
where *selection* is:



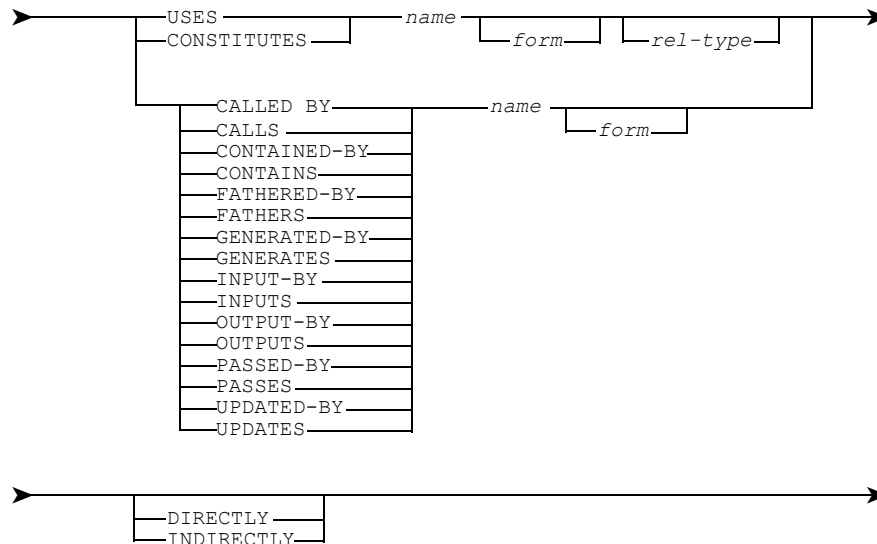
where:

list-name is the name of a KEPT-DATA list.

where *condition* is:



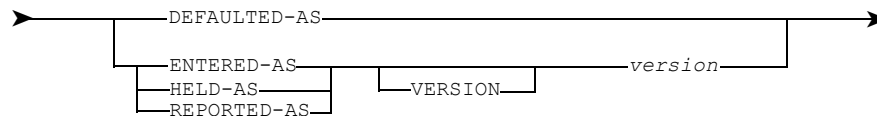
where *references-clause* is:



where:

name is the name of a repository member.

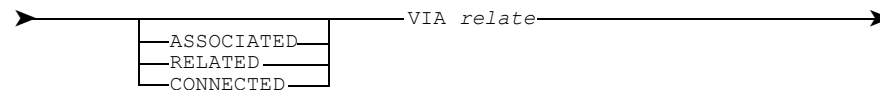
form is:



where:

version is an unsigned integer in the range 1 to 15.

rel-type is:

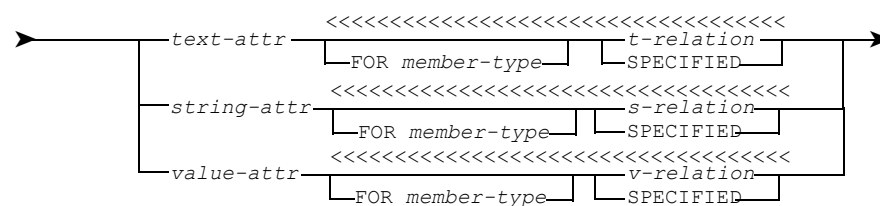


where:

relate is a relationship type or relationship class.

classification is a catalog classification.

attribute-specification is:

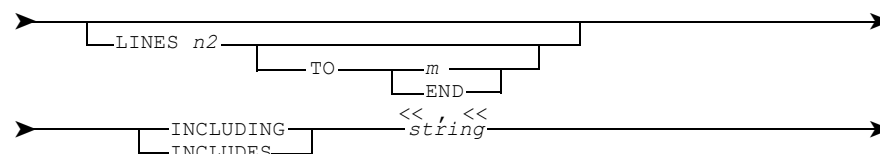


where:

text-attr is any attribute in a member definition that contains text, as shown on the output of the SHOW MEMBER-TYPE command.

member-type is as defined above.

t-relation is:



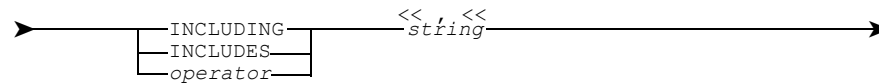
where:

$n2$ and m are unsigned integers in the range 1 to 32767.

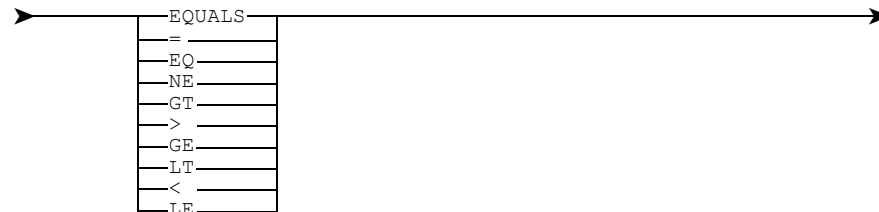
string is a character string of text.

string-attr is any attribute in a member definition that contains a string, as shown on the output of the SHOW MEMBER-TYPE command.

s-relation is:

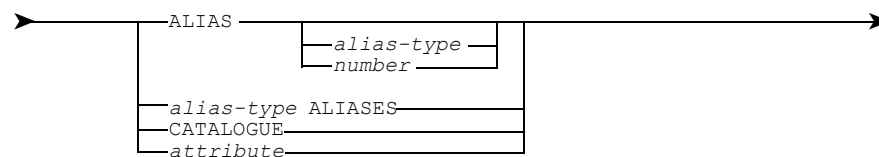


where *operator* is:



string is as defined above.

value-attr is:



where:

alias-type is as defined above.

number is as defined above.

attribute is any attribute in a member definition that contains neither text nor a string (see output of the `SHOW MEMBER-TYPE` command).

v-relation is:



where:

operator is as defined above.

value is any variable appropriate to the value-attribute.

owner-name is the name of an owner.

operator is as defined above.

n1 is an unsigned integer in the range 0 to 2,147,483,647.

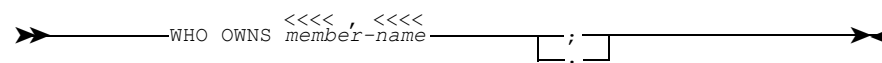
list-name is as defined above.

Note: _____

Classification, *owner-name*, and *string* may optionally be in quotes.

WHO

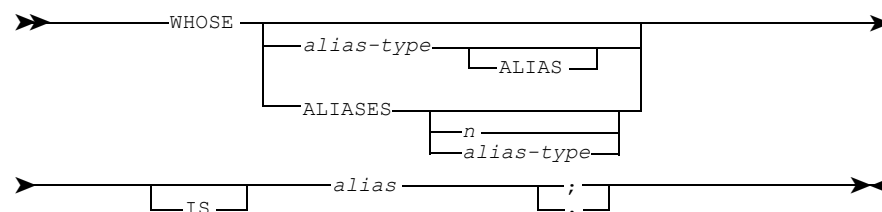
To ascertain the ownership of specified members of the repository.



where *member-name* is the name of a repository member.

WHOSE

To ascertain the encoded member(s) to which an alias relates.



where:

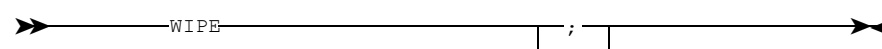
alias-type is one of the specific alias-type keywords available in your repository.

n can be any number from one to the maximum number of general aliases allowed in your repository.

alias is a string from 1 to 79 printable characters, being an alias recorded in the repository.

WIPE

To clear the current contents of the Scratchpad.



WORD Function

To return the n th word in a string.

➤—WORD(*string*, n)—➤

where:

string is a string.

n is an integer.

WORDINDX Function

To return the position of the n th word in a string.

➤—WORDINDX(*string*, n)—➤

where:

string is a string.

n is a positive integer.

WORDLEN Function

To return the position of the n th word in a string.

➤—WORDLEN(*string*, n)—➤

where:

string is a string.

n is a positive integer.

WORDS Function

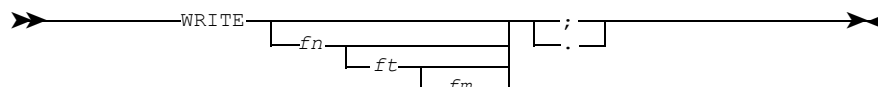
To return the number of words in a string.

➤—WORDS(*string*)—➤

where *string* is a string.

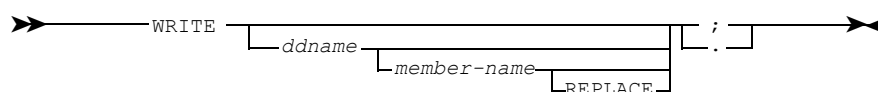
WRITE

To write the contents of any buffer to a CMS file.



where *fn*, *ft*, and *fm* are the file name, file type, and file mode of a CMS file.

To write to an MVS dataset.



where:

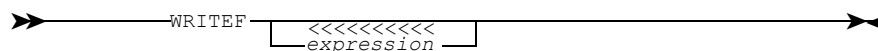
ddname is the ddname specified in a previously issued TSO ALLOCATE command.

member-name is the name of a PDS member to be written.

WRITEF Directive

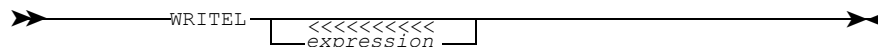
To evaluate all expressions, including functions and those involving arithmetic operations. Text can be output to:

- The primary/secondary output device
- User Member
- A sequential dataset
- A partitioned dataset



WRITEL Directive

To output a string to the Primary Output Device.



where *expression* is an expression subject to Limited Evaluation.

XQUIT

To close and delete an Update or Edit Buffer which has been amended, without filing the amended member.

➤ —XQUIT— □; □ ➤

ZONE

To define the columns to be searched when using the FIND, LOCATE, and CHANGE commands.

➤ —ZONE *mm nn*— □; □ ➤

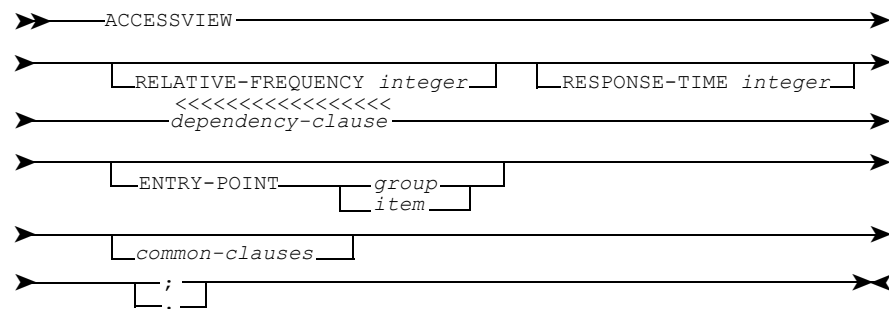
where *mm* is the first column and *nn* the last column of the zone.

2

Member Types

ACCESSVIEW

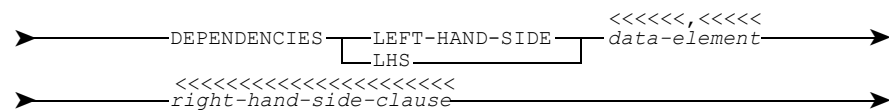
A set of dependencies which express an elementary function's requirements for accessing the values of data elements (items and certain groups) in the entity model.



where:

integer is an integer value of up to 18 digits, optionally preceded by a sign.

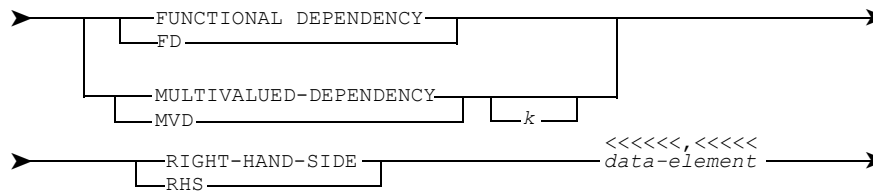
dependency-clause is:



where:

data-element is the name of a data element that is held in the modeling repository as an ITEM or GROUP member or is to be entered in the modeling repository as a dummy ITEM.

right-hand-side-clause is:



where:

k is an unsigned integer indicating the multiplicity for a multivalued dependency; that is, the average number of values (or sets of values) of the right-hand side of the dependency determined by a given value of the left-hand side. If not specified for a dependency, the default value of *k* is taken to be one. (The multiplicity of a functional dependency is automatically taken to be one.)

data-element is as defined above.

group is the name of a GROUP member.

item is the name of an ITEM member.

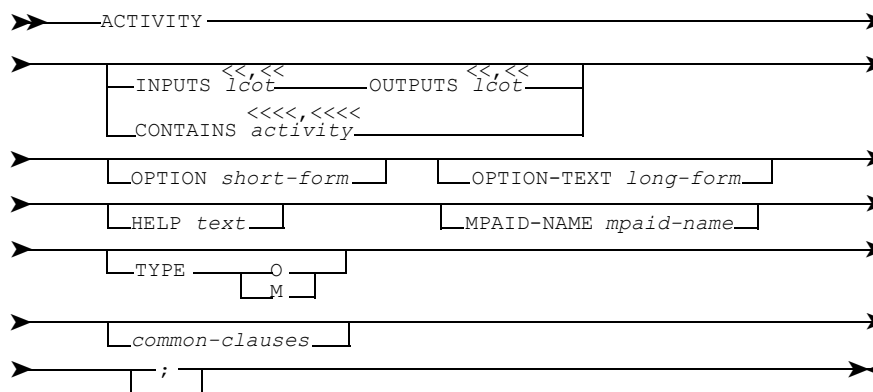
common-clauses are any of the clauses common to all member types.

Note:

The commas and delimiters shown in the above syntax are required when defining an ACCESSVIEW member via the command interface.

ACTIVITY

A stage within a phase which leads to a defined (intermediate) result in the form of one or more LIFE CYCLE OBJECT TYPES.



where:

lcot is the name of a LIFE-CYCLE-OBJECT-TYPE member.

activity is the name of an ACTIVITY member.

short-form is a delimited string of up to 32 characters.

long-form is a delimited string of up to 50 characters.

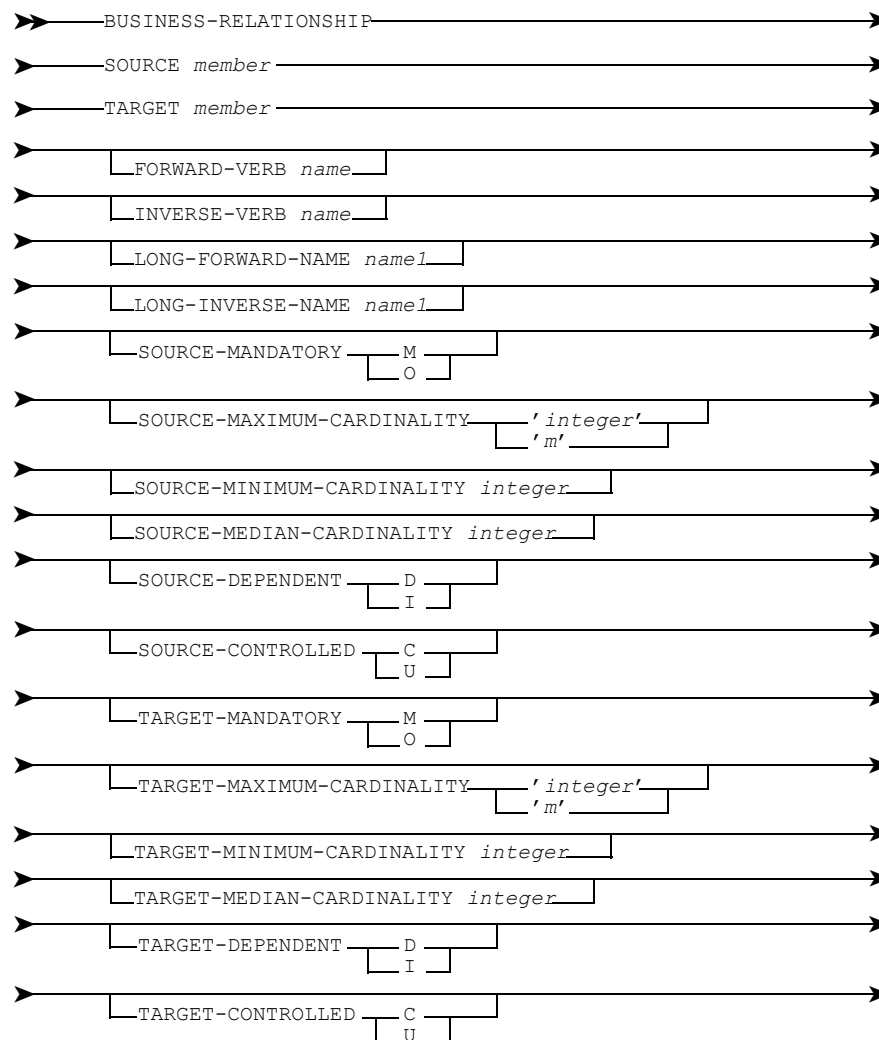
text is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

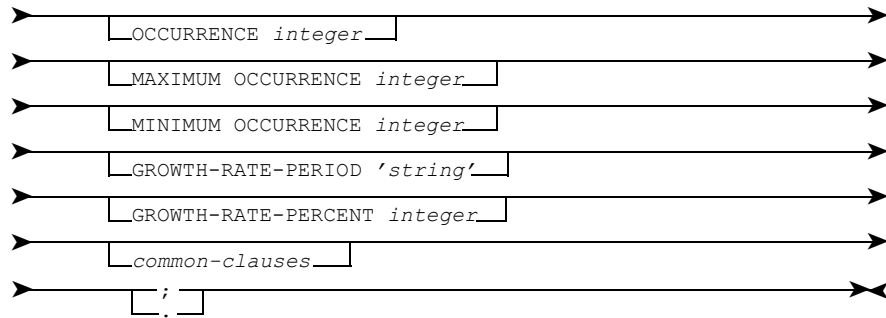
mpaid-name is a name of up to 10 characters.

common-clauses are any of the clauses common to all member types.

BUSINESS-RELATIONSHIP

To define a business relationship.





where:

member is a repository member name.

name is a string of up to 32 characters.

name1 is a string of up to 80 characters.

integer is an integer.

Note:

m is a literal, not a variable.

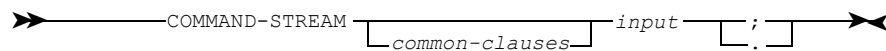
string is a string of up to 32 characters.

Note:

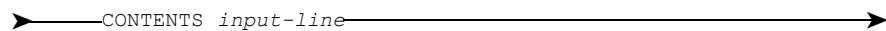
The delimiters shown in the above syntax are required when defining a BUSINESS-RELATIONSHIP member via the command interface.

COMMAND-STREAM

To store a stream of Manager Products commands.



where *input* is:



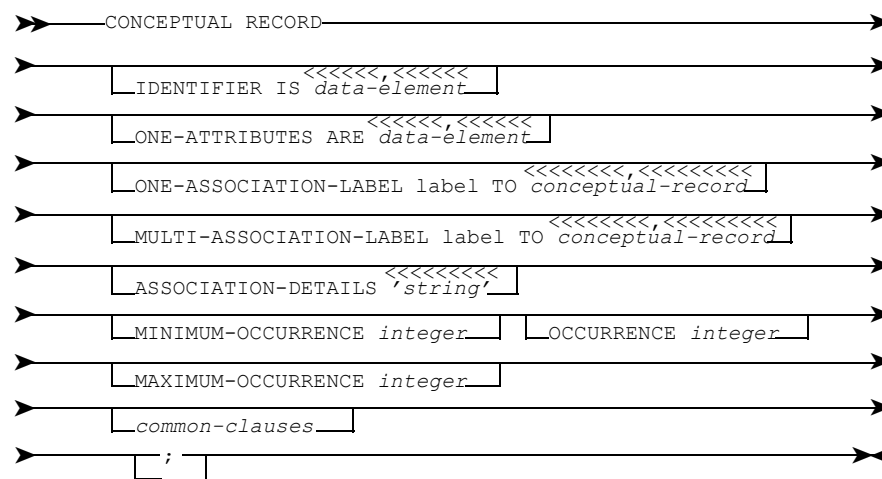
where:

input-line is any Manager Products command (except a PERFORM command, unless the ASG-ControlManager User Defined Commands facility is installed). The input line may contain an asterisk (or other character determined by a SET PERFORM-CHARACTER command) where a member-name, alias, catalog classification, or (if the User Defined Syntax facility is installed) a user-defined indexed attribute would normally occur. If the input line would otherwise have a terminator in the first character position, that terminator should be placed in the second character position with % (or another character determined by the SET ESCAPE-CHARACTER command) in the first character position.

common-clauses are any of the clauses common to all member types.

CONCEPTUAL-RECORD

A primary key and its descriptive elements.



where:

data-element is the name of a GROUP or ITEM member.

label is a text string of up to 32 characters.

conceptual-record is the name of a CONCEPTUAL-RECORD member.

'string' is a string of up to 32 characters.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

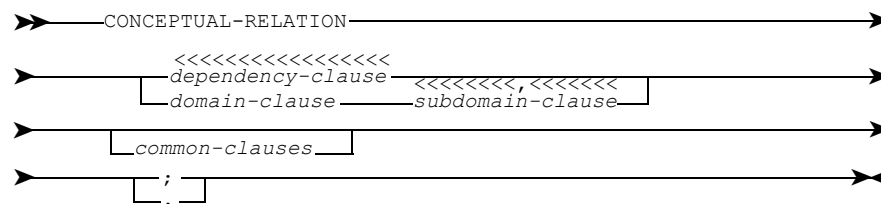
common-clauses are any of the clauses common to all member types.

Note:

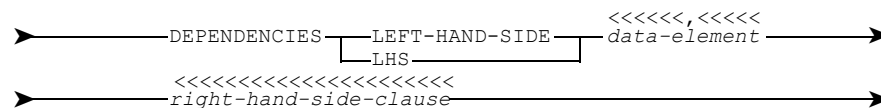
The delimiters shown in the above syntax are required when defining a CONCEPTUAL-RECORD member via the command interface.

CONCEPTUAL-RELATION

A logical description of a set of data elements in at least first normal form.



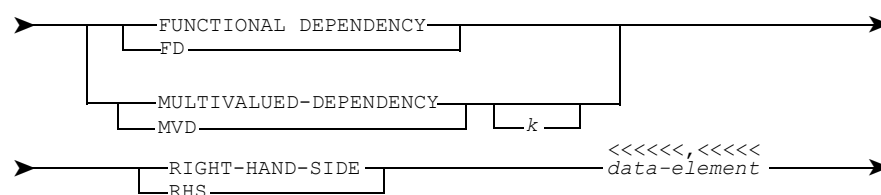
where *dependency-clause* is:



where:

data-element is the name of a data element that is held in the modeling repository as an ITEM or GROUP member or is to be entered in the modeling repository as a dummy ITEM.

right-hand-side-clause is:



where:

k is an unsigned integer indicating the multiplicity for a multivalued dependency; that is, the average number of values (or sets of values) of the right-hand side of the dependency determined by a given value of the left-hand side. If not specified for a dependency, the default value of k is taken to be one. (The multiplicity of a functional dependency is automatically taken to be one.)

data-element is as defined above.

domain-clause is:

► — DOMAIN IS ^{<<<<<, <<<<<<}
data-element —►

where:

data-element is as defined above.

subdomain-clause is:

► — SUB-DOMAIN IS ^{<<<<<, <<<<<<}
data-element —►

where:

data-element is as defined above.

common-clauses are any of the clauses common to all member types.

CRITICAL-SUCCESS-FACTOR

One of the limited number of areas of business activity whose results, if they are satisfactory, will ensure successful competitive performance for the organizational unit and the attainment of its goals.

► — CRITICAL-SUCCESS-FACTOR
CSF —►

► — PRIME-MEASURE ^{<<<<<<<, <<<<<<<}
group item —►

► — CORPORATE-RATING *integer* —►

► — common-clauses —►

► — ; —►

where:

group is the name of a GROUP member.

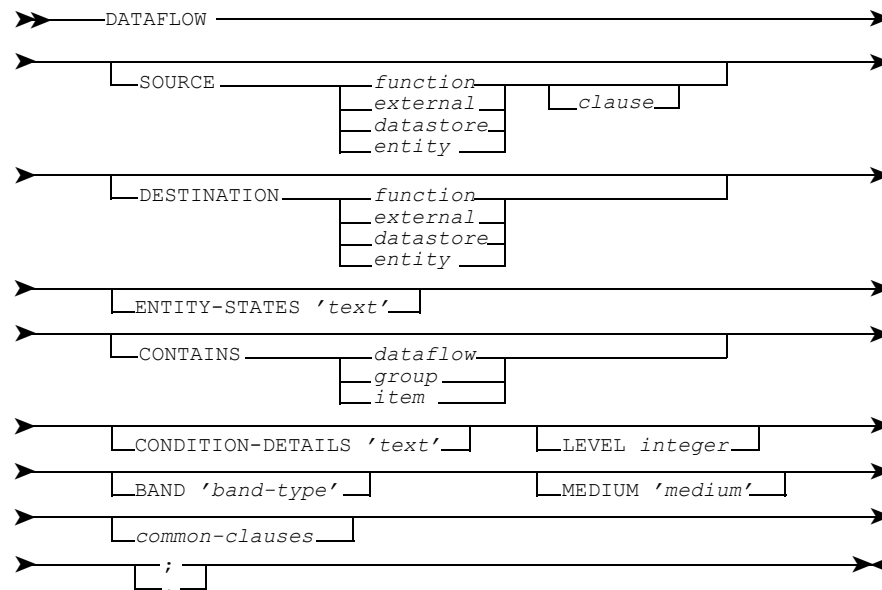
item is the name of an ITEM member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

common-clauses are any of the clauses common to all member types.

DATAFLOW

A collection of several data elements being jointly transmitted from a source to a destination.



where:

function is the name of a FUNCTION member.

external is the name of an EXTERNAL member.

datastore is the name of a DATASTORE member.

entity is the name of an ENTITY member.

clause is:



where:

condition-expression is a string of up to 0 characters, including delimiters.

'text' is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

dataflow is the name of a DATAFLOW member.

group is the name of GROUP member.

item is the name of an ITEM member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

'*medium*' may contain a text string of up to 32767 delimited character strings, each up to 252 characters long.

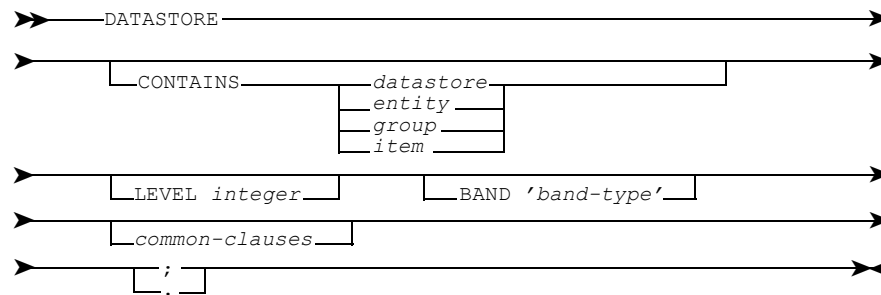
Note:

The delimiters shown in the above syntax are required when defining a DATAFLOW member via the command interface.

common-clauses are any of the clauses common to all member types.

DATASTORE

A logical storage place for data consisting of related entities, groups, items, and possibly other datastores.



where:

datastore is the name of a DATASTORE member.

entity is the name of an ENTITY member.

group is the name of a GROUP member.

item is the name of an ITEM member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

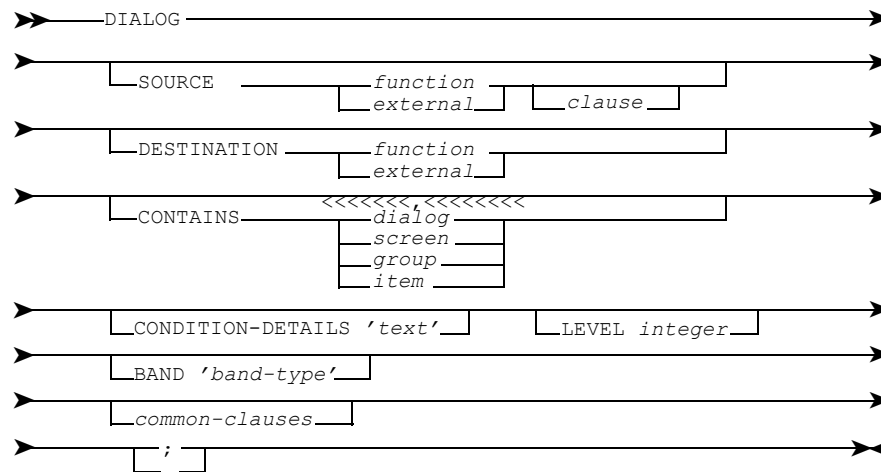
Note:

The delimiters shown in the above syntax are required when defining a DATASTORE member via the command interface.

common-clauses are any of the clauses common to all member types.

DIALOG

A series of interactive exchanges, each comprising of a display of information by the system followed by a user response, as defined by one or more functions.



where:

function is the name of a FUNCTION member.

external is the name of an EXTERNAL member.

clause is:



where:

condition-expression is a string of up to 79 characters, including delimiters.

dialog is the name of a DIALOG member.

screen is the name of a SCREEN member.

group is the name of a GROUP member.

item is the name of an ITEM member.

'*text*' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is at next string of up to 78 characters, including delimiters.

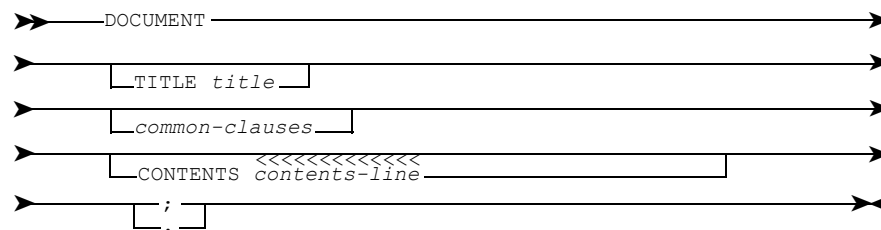
Note: _____

The delimiters shown in the above syntax are required when defining a DIALOG member via the command interface.

common-clauses are any of the clauses common to all member types.

DOCUMENT

To define a repository document.



where:

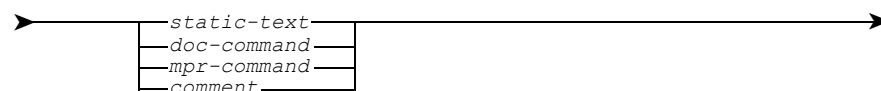
title is a text string of up to 32767 delimited character strings, each up to 252 characters long. It is used as an entry in the table of contents when the document is incorporated into another document.

common-clauses are any of the clauses that are common to all member types.

Note: _____

The NOTE clause has special significance in a DOCUMENT member; it is used as an introductory heading when the document is incorporated into another document.

contents-line is a single line of text which may be any of these:



where:

static-text is a linen of text that is reproduced without change in the output document.

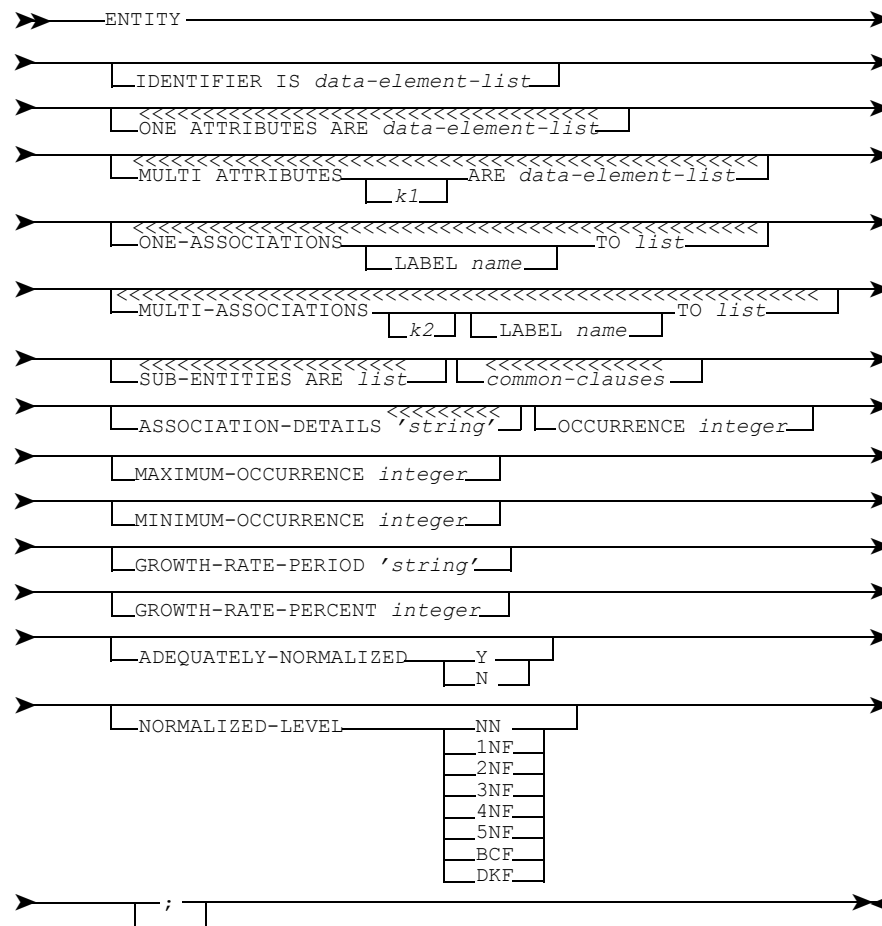
doc-command is a DOCUMENT command, preceded by an appropriate escape sequence and followed by a semicolon terminator.

mpr-command is a Manager Products command, preceded by an appropriate escape sequence and followed by a semicolon terminator.

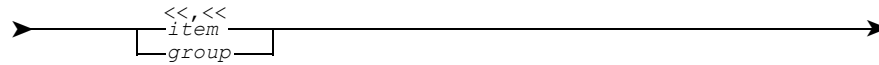
comment is a line of text that is not reproduced in the output document, preceded by an appropriate escape sequence.

ENTITY

To define a business entity.



where *data-element list* is:



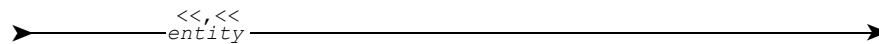
where:

item is the name of an ITEM.

group is the name of a GROUP.

k1 is an unsigned integer indicating the multiplicity of the specified set of one or more multi-attributes, that is, the average number of values of the listed data element (or set of data elements) determined by a single value of the identifier of the given entity.

name is a text string of up to 32 characters.



where:

entity is the name of an ENTITY.

k2 is an unsigned integer indicating the multiplicity of the specified multi-association, that is, the expected number of instances of the listed target entity (or set of target entities) determined by a single instance of the given entity.

common-clauses are any of the clauses common to all member types.

'*string*' is a string of up to 32 characters.

integer is an integer.

Note:

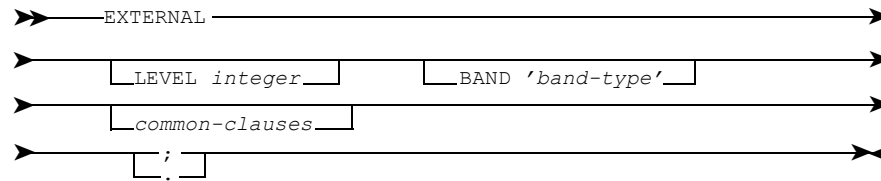
The delimiters shown in the above syntax are required when defining an ENTITY member via the command interface.

EXECUTIVE-ROUTINE

See [Chapter 3, "Procedures Language," on page 295](#) for further details of the EXECUTIVE-ROUTINE member type.

EXTERNAL

An entity or process which lies outside the area under study and is the source or destination of one or more dataflows.



where:

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

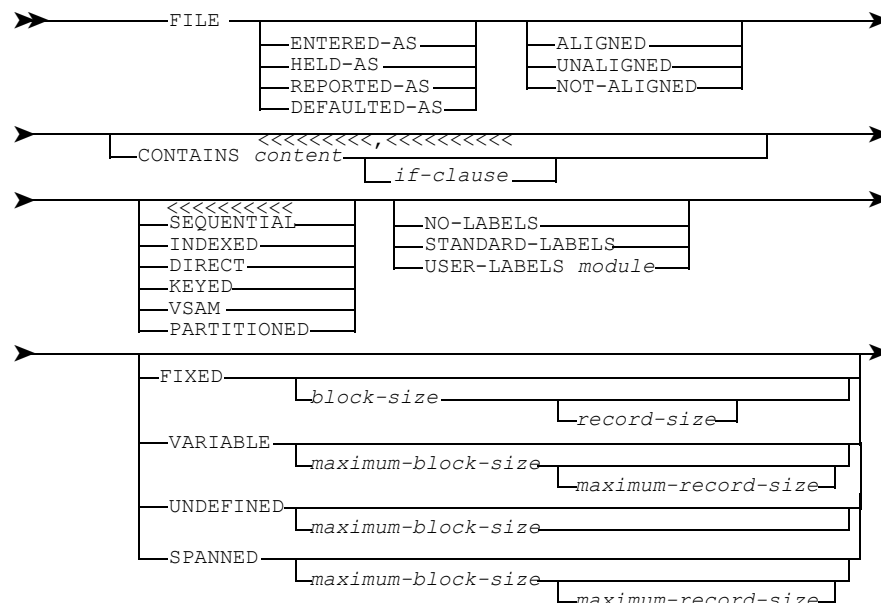
Note:

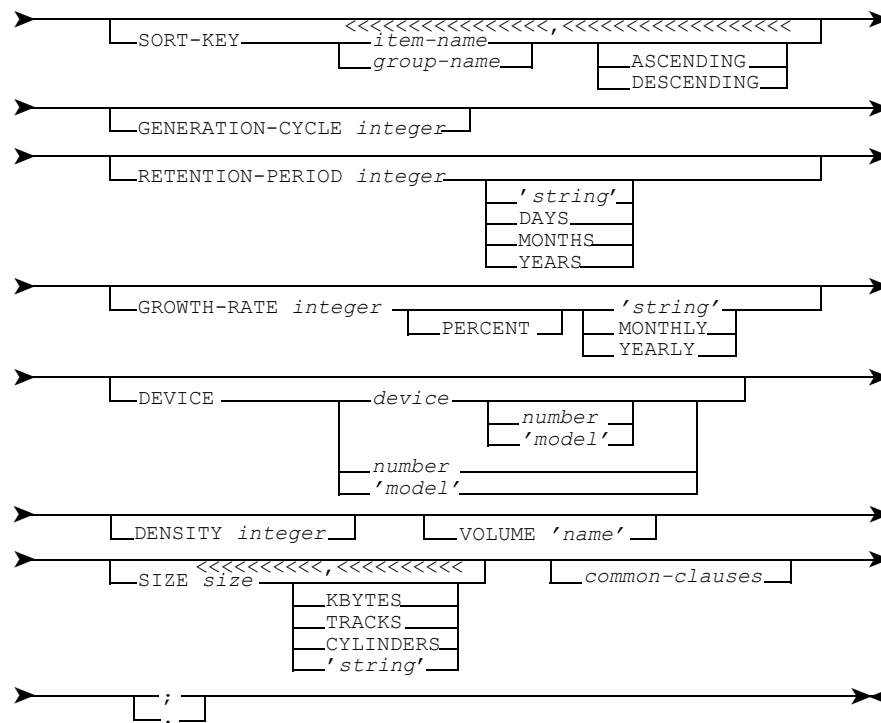
The delimiters shown in the above syntax are required when defining an EXTERNAL member via the command interface.

common-clauses are any of the clauses common to all member types.

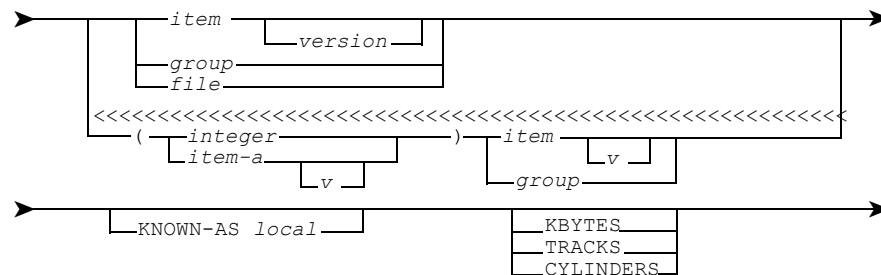
FILE

A set of related records treated as a unit.





where *content* is:



where:

item is the name of an ITEM member.

v is an unsigned integer in the range 1 to 15.

group is the name of a GROUP member.

file is the name of a FILE member.

integer is an unsigned integer not greater than 32767.

item-a is the name of an ITEM.

local is a name conforming to the rules for member names.

[illegible]

item-b is the name of an ITEM to be compared with *item-c* or a literal.

item-c is the name of an ITEM to be compared with *item-b*.

EQ (or $=$) means equal to.

NE means not equal to.

GT (or $>$) means greater than.

GE means greater than or equal to.

LT (or $<$) means less than.

LE means less than or equal to.

literal is the literal being compared with item-b.

content is as defined above.

module is the name of MODULE.

block-size is an unsigned integer, being the number of characters per block.

record-size is an unsigned integer, being the number of characters per record.

maximum-block-size is an unsigned integer not greater than 32767.

maximum-record-size is an unsigned integer not greater than 32767.

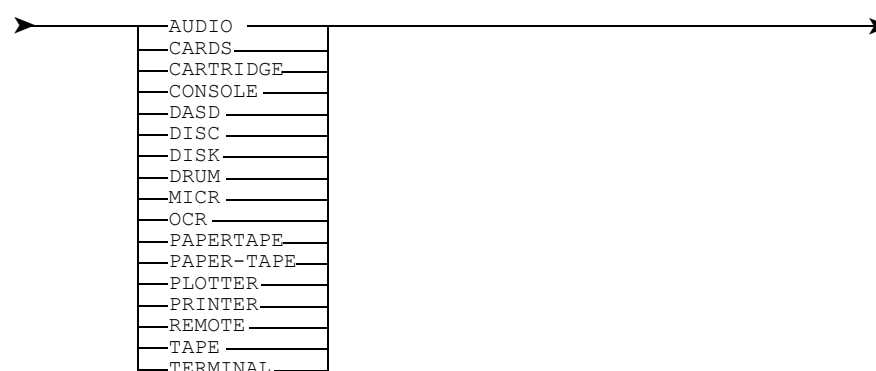
item is as defined above.

group is as defined above.

integer is as defined above.

'*string*' is a character string of not more than 256 characters.

device is:



number is a manufacturer's type number or model number, with a maximum of eight digits.

'*model*' is a character string of not more than eight characters, being a manufacturer's model identifier. The string may include blank spaces.

'*name*' is an unsigned integer of between 1 and 10 digits, with a maximum value of 2147483647.

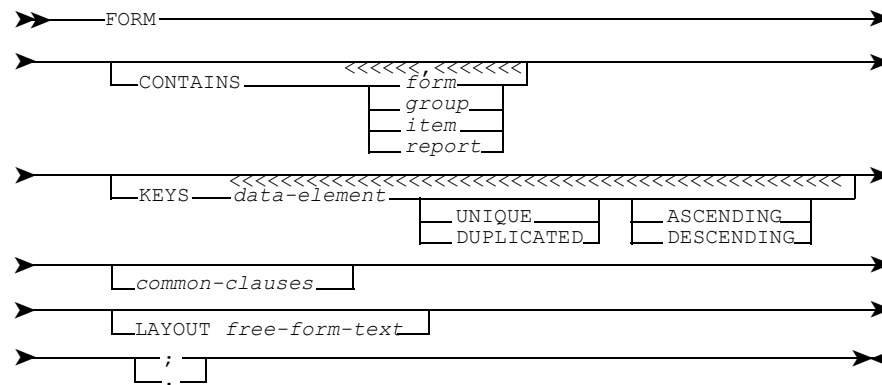
common-clauses are any of the clauses common to all member types.

FMT-SCREEN

See *ASG-Manager Products Quick Reference Administration* for further details of the FMT-SCREEN member type.

FORM

A formatted screen or document used for the input of information into a computer system.



where:

report is the name of a REPORT member.

form is the name of a FORM member.

group is the name of a GROUP member.

item is the name of an ITEM member.

data-element is the name of a DATAFLOW, ITEM, or GROUP member.

common-clauses are any of the clauses common to all member types.

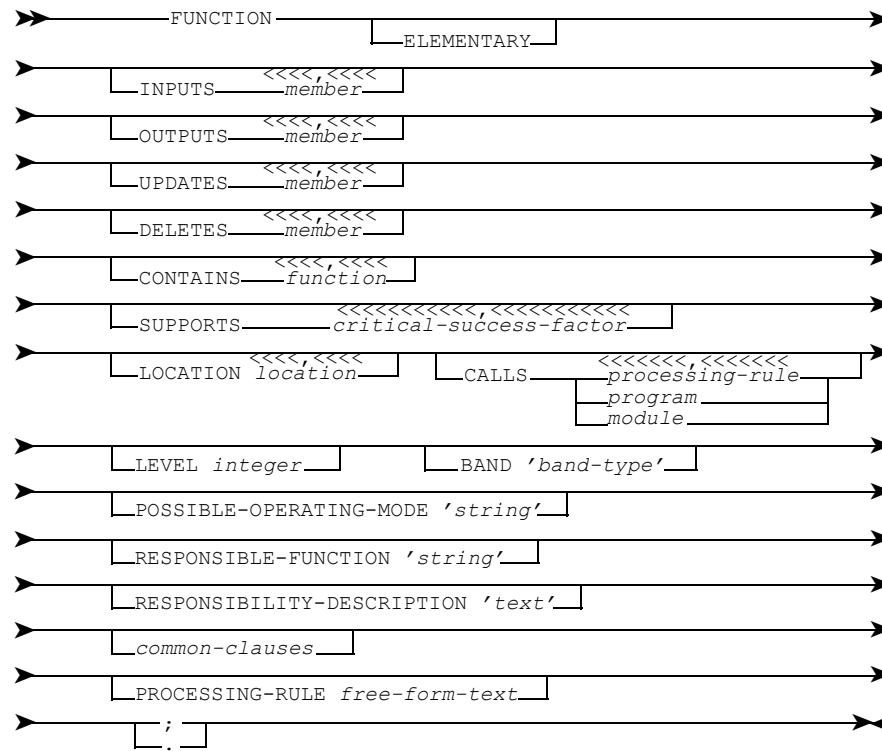
free-form-text is a maximum of 32767 lines of text, each containing a maximum of 256 characters.

FORMAT

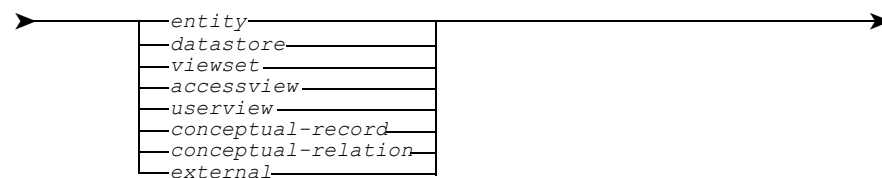
See *ASG-Manager Products Quick Reference: Admin* for further details of the FORMAT member type.

FUNCTION

A process applied to input information to produce output information, according to established rules.



where *member* is:



function is the name of a FUNCTION member.

critical-success-factor is the name of a CRITICAL-SUCCESS-FACTOR member.

location is the name of a LOCATION member.

processing-rule is the name of a PROCESSING-RULE member.

program is the name of a PROGRAM member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

'*string*' is a character string of not more than 256 characters.

'*text*' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

common-clauses are nay of the clauses that are common to all member types.

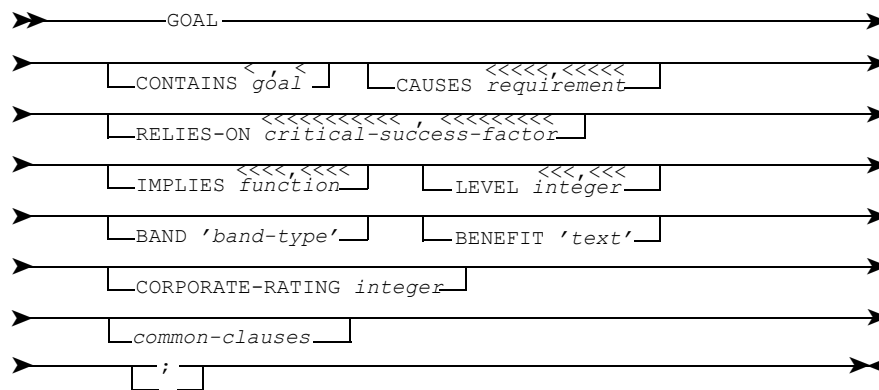
free-form-text is a maximum of 32767 lines of text, each line containing a maximum of 248 characters.

Note: _____

The delimiters shown in the above syntax are required when defining a FUNCTION member via the command interface.

GOAL

The state or position that an organizational unit wishes to attain.



where:

goal is the name of a GOAL member.

requirement is the name of a REQUIREMENT member.

critical-success-factor is the name of a CRITICAL-SUCCESS-Factor (CSF) member.

function is the name of a FUNCTION member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

'*text*' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

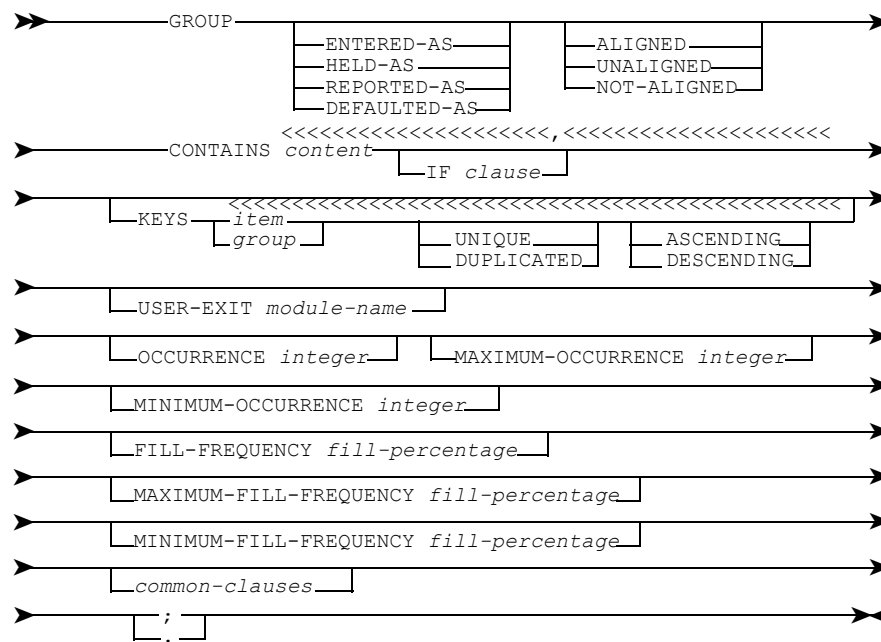
Note:

The delimiters shown in the above syntax are required when defining a GOAL member via the command interface.

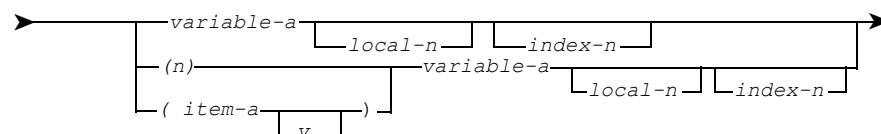
common-clauses are any of the clauses common to all member types.

GROUP

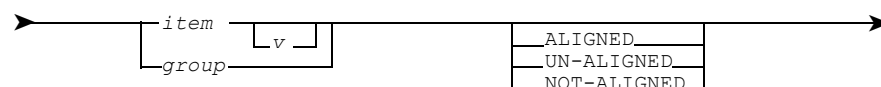
An ordered combination of items and/or other groups.



where *content* is:



where *variable-a* is:



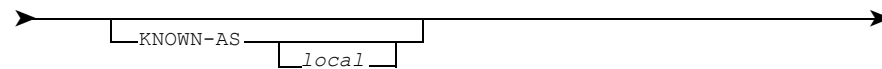
where:

item is the name of an ITEM member.

v is an unsigned integer in the range of 1 to 15. *v* is not allowed with the keyword KEYS.

group is the name of a GROUP member.

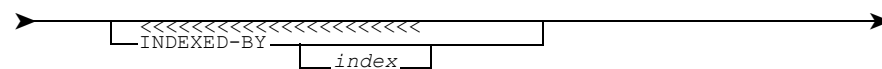
local-n is:



where:

local is an alternative name for an ITEM or GROUP.

index-n is:



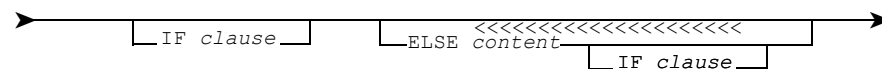
where:

index is the name used as the index name when COBOL data descriptions are generated by Source Language Generation.

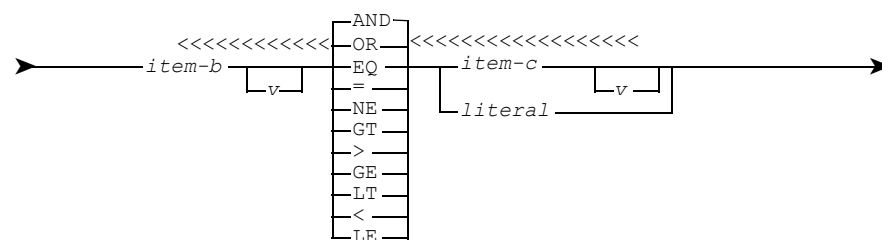
n is an unsigned integer from 1 to 999,999,999.

item-a is the name of an ITEM member.

if-clause is:



where *clause* is:



where:

item-b is the name of an ITEM to be compared with *item-c* or a literal.

v is as defined above.

item-c is the name of an ITEM to be compared with *item-b*.

EQ (or =) means equal to.

NE means not equal to.

GT (or >) means greater than.

GE means greater than or equal to.

LT (or <) means less than.

LE means less than or equal to.

literal is the literal being compared with item-b.

content is as defined above.

group is as defined above.

item is as defined above.

module is the name of a MODULE member.

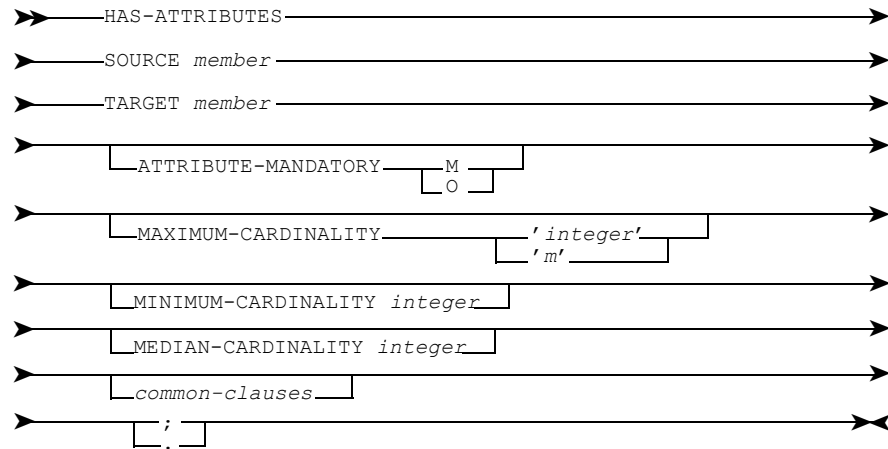
integer is an integer value of up to 18 digits, optionally preceded by a sign.

fill-percentage is an integer in the range 1 to 100.

common-clauses are any of the clauses common to all member types.

HAS-ATTRIBUTES

To associate an attribute with a business entity or business relationship.



where:

member is a repository member name.

integer is a positive integer.

m is a literal, not a variable.

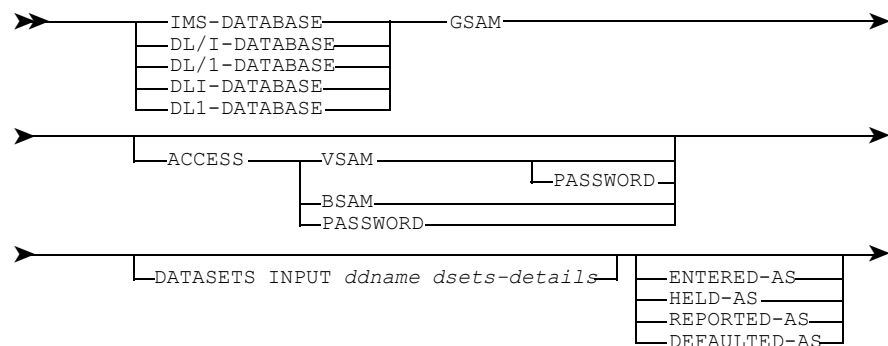
common-clauses are any of the clauses common to all member types.

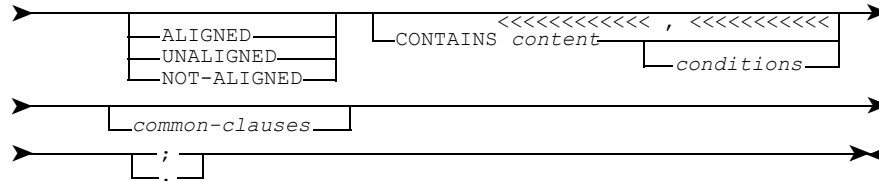
Note:

The delimiters shown in the above syntax are required when defining a HAS-ATTRIBUTES member via the command interface.

IMS-DATABASE GSAM

To define IMS (DL/I) databases using the generalized sequential access method.

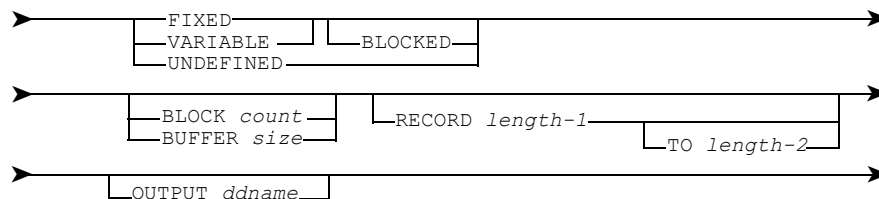




where:

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the database dataset.

dsets-details are:



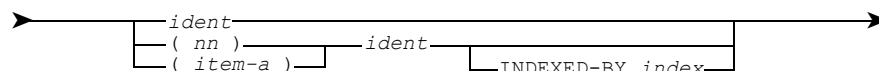
where:

count is an unsigned, non-zero integer, being the number of logical records per physical block.

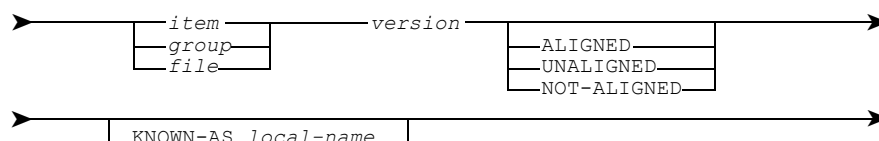
size is an unsigned, non-zero integer, being the number of bytes per physical block or control interval.

$length-1, length-2$ are non-zero integers.

content is:



where *ident* is:



where:

item is the name of an ITEM member.

group is the name of a GROUP member.

file is the name of a FILE member.

version is an unsigned integer in the range 1 to 15.

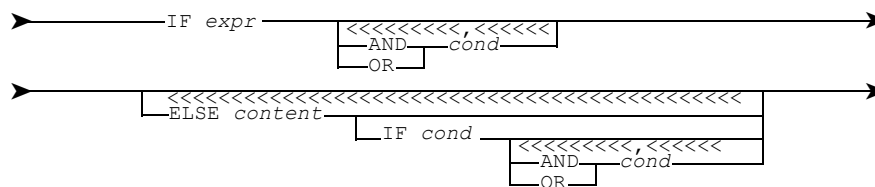
local-name is a name, conforming to the rules for member names.

nn is an unsigned integer of from 1 to 18 digits, being the number of times item or group occurs in the array.

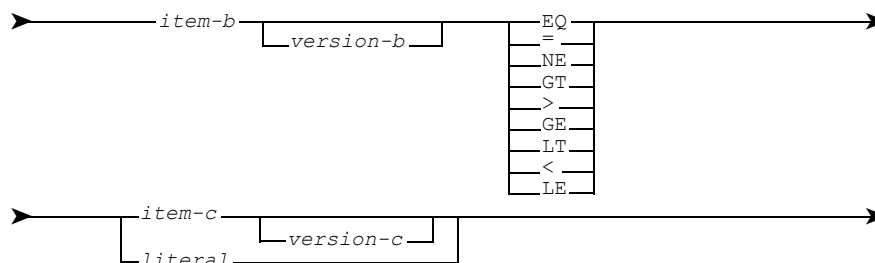
item-a is the name of an ITEM.

index is a name, conforming to the rules for member names, that is to be used as the index name when COBOL data descriptions are generated by the Source Language Generation facility.

conditions are:



where *cond* is:



where:

literal is a literal comparand.

item-b is the name of the ITEM whose contents are to be compared with the comparand.

version-b is an unsigned integer in the range 1 to 15.

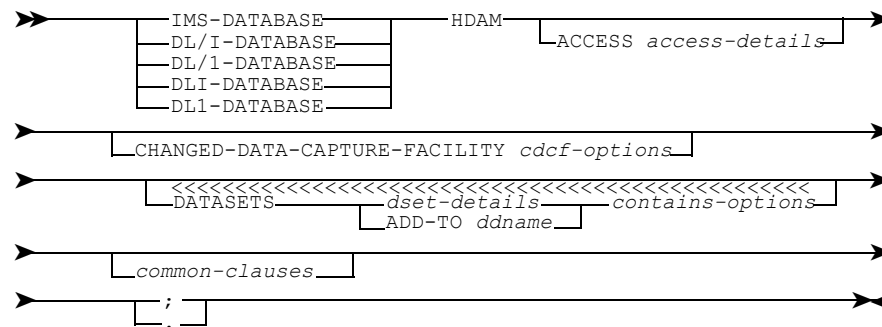
item-c is the name of the ITEM whose contents are the comparand.

version-c is an unsigned integer in the range 1 to 15.

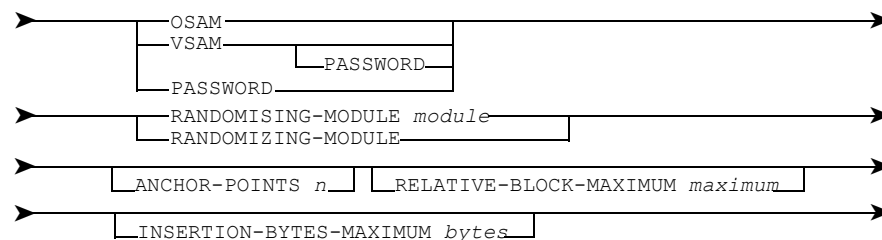
common-clauses are as defined in [Appendix B, "Common Clauses," on page 305](#).

IMS -DATABASE HDAM

To define IMS (DL/I) databases using the hierarchical direct access method.



where *access-details* are:



where:

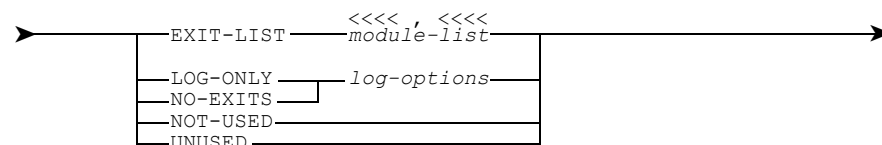
module is the name of a MODULE member.

n is an unsigned integer in the range 1 to 255, being the number of root anchor points required in each control interval or block.

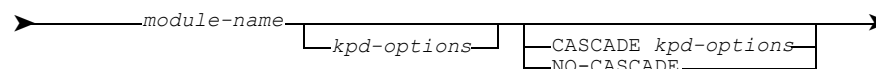
maximum is an unsigned integer in the range 1 to 16777215, being the maximum block number to be produced by the randomizing module.

bytes is an unsigned integer in the range 1 to 16777215, being the maximum number of bytes to be inserted into the root addressable area.

cdcf-options are:



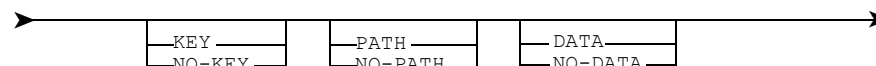
where *module-list* is:



where:

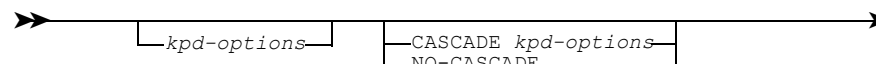
module-name is the name of a MODULE or PROGRAM member.

kpd-options are:

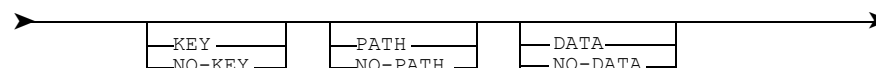


db-version is a delimited string of up to 255 characters.

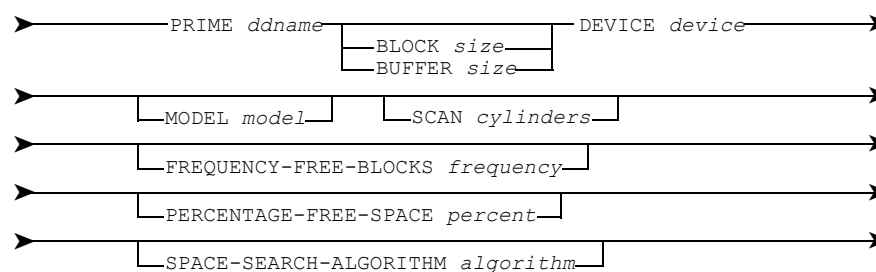
log-options are:



where *kpd-options* are:



dsets-detail are:



where:

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the physical file.

size is the number of bytes required per physical block or control interval.

device is one of the keywords or numbers from the list:

```
DRUM 2311    3310    3350
CELL  2314    3330    3370
2301   2319    3340    3375
2305   2321    3344    3380
3390
```

From IMS Version 4 onwards, this clause is purely for documentation purposes.

algorithm is an unsigned integer in the range 0 to 2.

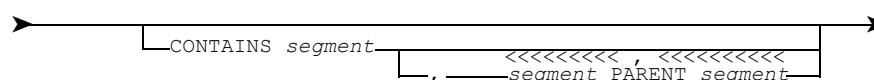
model is an integer, 1 or 2 if device is 2305, or 1 or 11 if device is 3330. From IMS version 4 onwards, this clause is purely for documentation purposes.

cylinders is an unsigned integer in the range 0 to 255.

frequency is an unsigned integer in the range 2 to 100, or is 0.

percent is an unsigned integer in the range 0 to 99.

contains-options are:



where:

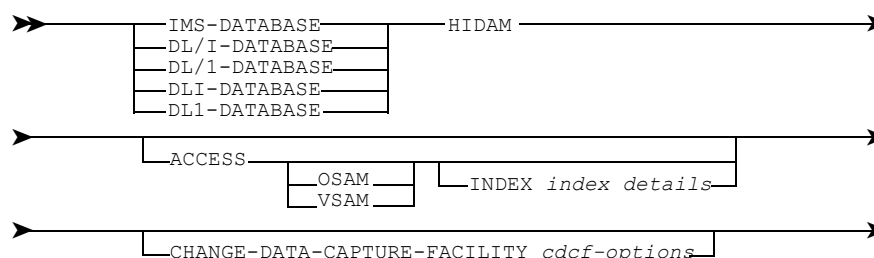
segment is the name of any physical segment.

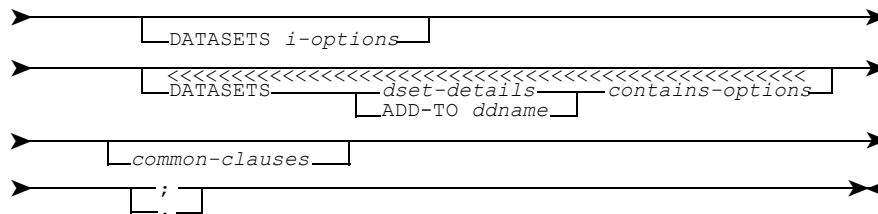
ddname is as defined above.

common-clauses are as defined in [Appendix B, "Common Clauses," on page 305](#).

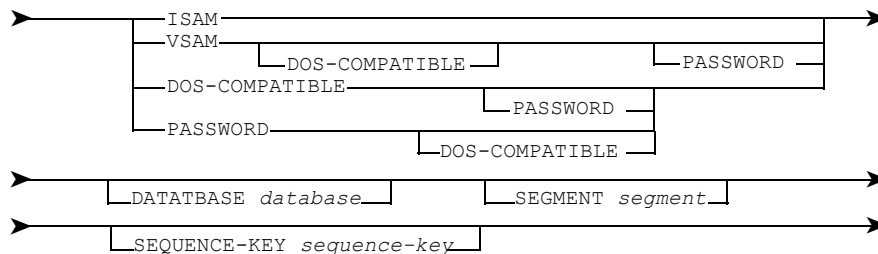
IMS-DATABASE HIDAM

To define IMS (DL/I) databases, including primary index databases, using the hierarchical indexed direct access method.





where *index-details* are:



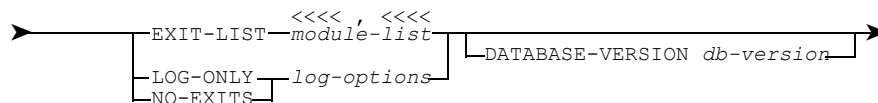
where:

database is 1 to 8 alphanumeric characters, being the IMS name of the primary index database associated with this HIDAM database.

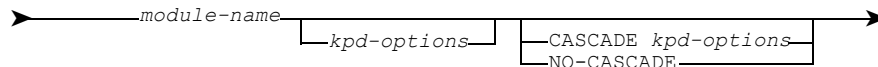
segment is 1 to 6 alphanumeric characters, being the IMS name of the primary index segment associated with this HIDAM database.

sequence-key is 1 to 8 alphanumeric characters, being the IMS sequence key name of the primary index database associated with this HIDAM database.

cdcf-options are:



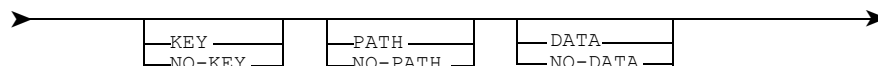
where *module-list* is:



where:

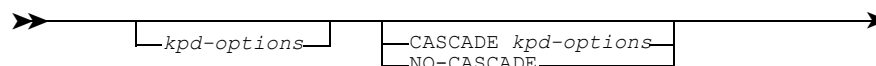
module-name is the name of a MODULE or PROGRAM member.

kpd-options are:

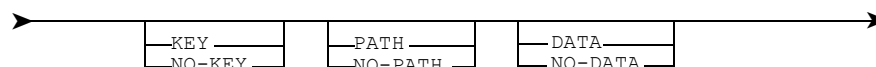


db-version is a delimited string of up to 255 characters.

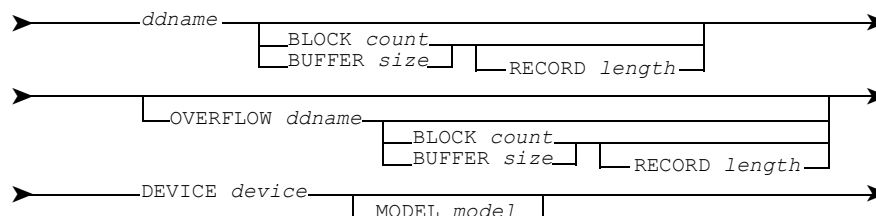
log-options are:



where *kpd-options* are:



i-options are:



where:

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the physical file.

count, *size*, and *length* are all unsigned non-zero integers.

device is one of the keywords or numbers from the list:

DRUM	2311	3310	3350
CELL	2314	3330	3370
2301	2319	3340	3375
2305	2321	3344	3380
3390			

From IMS version 4 onwards, this clause is purely for documentation.

model is an integer, 1 or 2 if device is 2305, or 1 or 11 if device is 3330. From IMS version 4 onwards, this clause is purely for documentation.

dset-details are:

where:

cylinders is an unsigned integer in the range 0 to 255.

percent is an unsigned integer in the range 0 to 99.

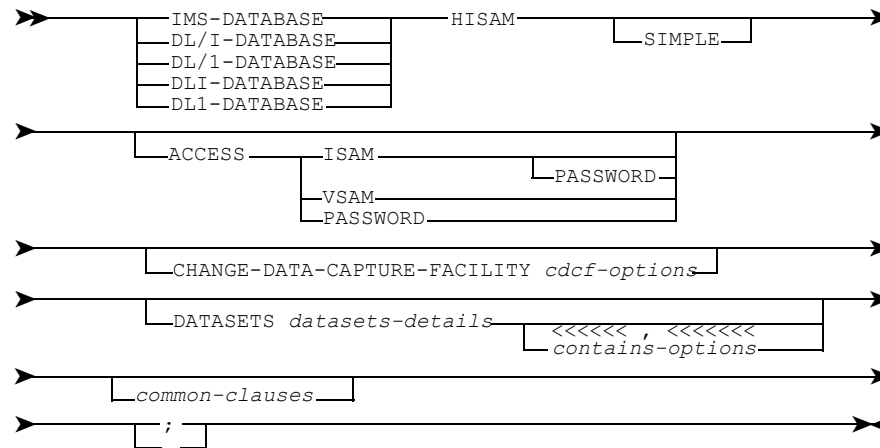
contains-options are:

where:

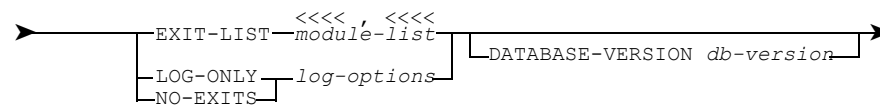
common-clauses are as defined in [Appendix B, "Common Clauses," on page 305](#).

IMS-DATABASE HISAM

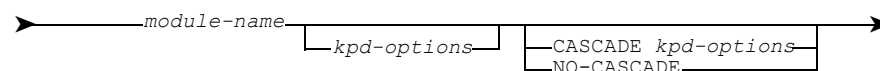
To define IMS (DL/I) databases using the hierarchical indexed sequential access method or its simple variant SHISAM.



where *cdcfc-options* are:



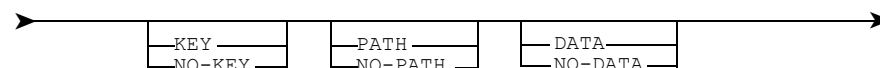
where *module-list* is:



where:

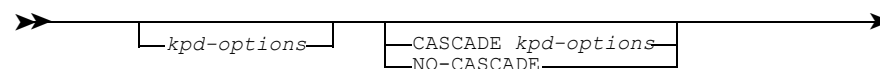
module-name is the name of a MODULE or PROGRAM member.

kpd-options are:

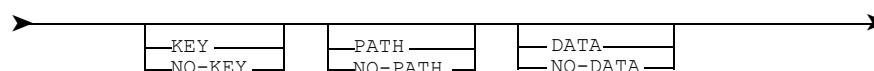


db-version is a delimited string of up to 255 characters.

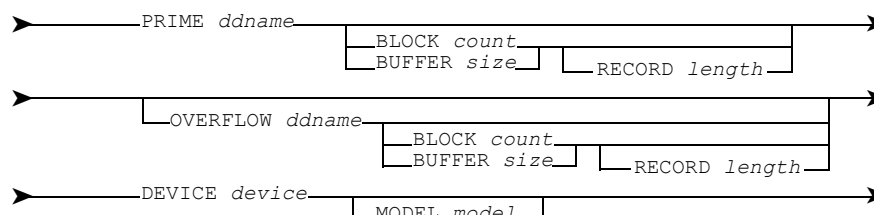
log-options are:



where *kpd-options* are:



dataset-details are:



where:

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the database dataset.

count is an unsigned, non-zero integer, being the number of logical records per physical block.

size is an unsigned, non-zero integer, being the number of bytes required per physical block or control interval.

length is an unsigned non-zero integer, being the maximum length (in bytes) of a logical record. If VSAM is the operating system access method, length must be an even value.

device is one of the keywords or numbers from the list:

DRUM	2311	3310	3350
CELL	2314	3330	3370
2301	2319	3340	3375
2305	2321	3344	3380
3390			

From IMS version 4 onwards, this clause is purely for documentation purposes.

model is an integer, 1 or 2 if device is 2305, or 1 or 11 if device is 3330. From IMS version 4 onwards, this clause is purely for documentation purposes.

device is one of the keywords or numbers from the list:

DRUM	2311	3310	3350
CELL	2314	3330	3370
2301	2319	3340	3375
2305	2321	3344	3380
3390			

From IMS version 4 onwards, this clause is purely for documentation purposes.

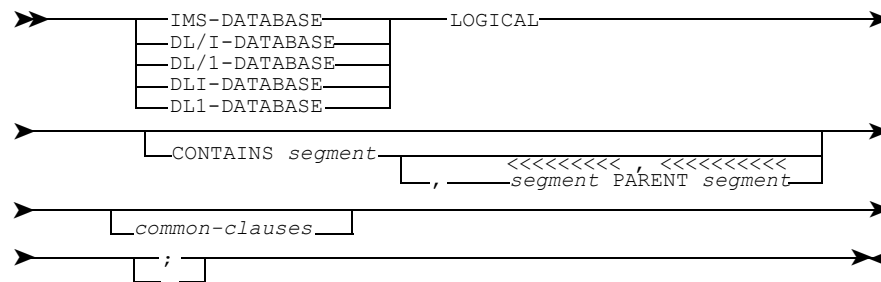
model is an integer, 1 or 2 if device is 2305, or 1 or 11 if device is 3330. From IMS version 4 onwards, this clause is purely for documentation purposes.

segment is the name of any physical segment.

common-clauses are as defined in [Appendix B, "Common Clauses,"](#) on page 305.

IMS-DATABASE LOGICAL

To define IMS (DL/I) logical databases.



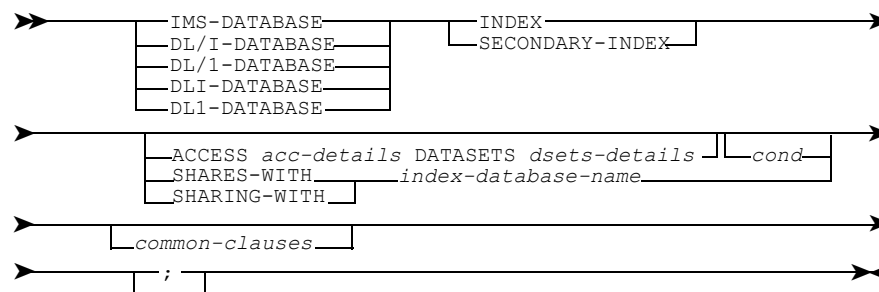
where:

segment is the name of a logical or physical segment.

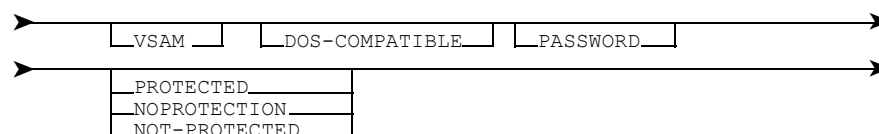
common-clauses are as defined in [Appendix B, "Common Clauses,"](#) on page 305.

IMS-DATABASE SECONDARY-INDEX

To define IMS (DL/I) secondary index databases.

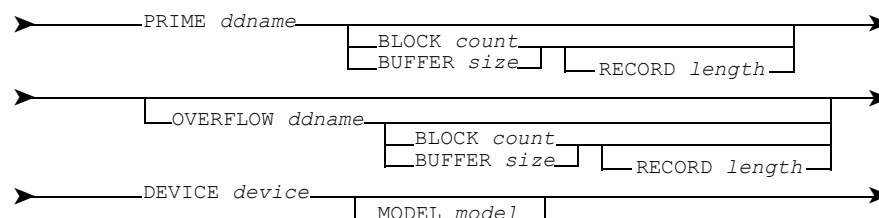


where *acc-details* are:



index-database-name is the name of another IMS (DL/I) SECONDARY-INDEX database

dsets-details are:



where:

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the physical file.

count, *size*, and *length* are all unsigned non-zero integers.

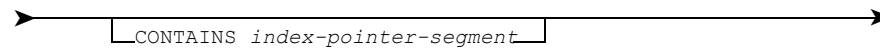
device is one of the keywords or numbers from the list:

DRUM	2311	3310	3350
CELL	2314	3330	3370
2301	2319	3340	3375
2305	2321	3344	3380
3390			

From IMS Version 4 onwards, this clause is purely documentary.

model is an integer, 1 or 2 if device is 2305, or 1 or 11 if device is 3330. From IMS version 4 onwards, this clause is purely documentary.

cont is:



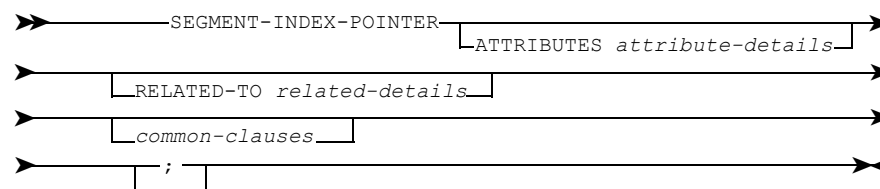
where:

index-pointer-segment is an INDEX-POINTER-SEGMENT member.

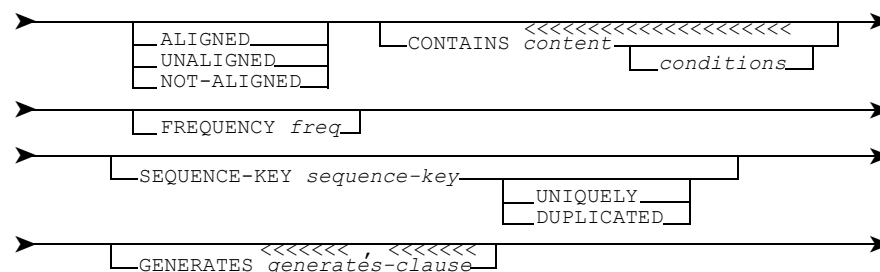
common-clauses are as defined in [Appendix B, "Common Clauses,"](#) on page 305.

INDEX-POINTER-SEGMENT

To record information relevant to the secondary indexing of IMS databases.



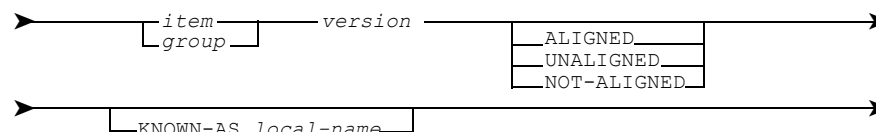
where *attribute-details* are:



where *content* is:



where *indent* is:



where:

item is the name of an ITEM member.

group is the name of a GROUP member.

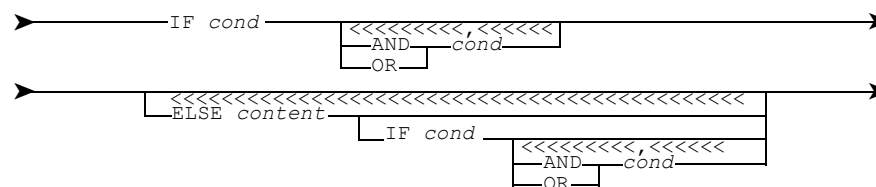
version is an unsigned integer in the range 1 to 15.

local-name is a name conforming to the rules for member names.

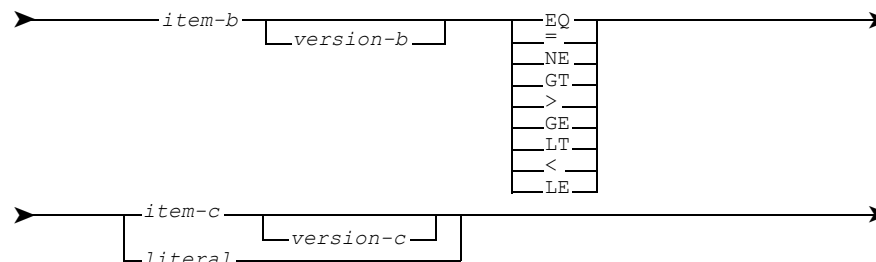
nn is an unsigned integer of from 1 to 18 digits, being the number of times item or group occurs in the array.

index is a name, conforming to the rules for member names, that is to be used as the index name when COBOL data descriptions are generated by the Source Language Generation facility.

conditions are:



cond is:



where:

item-b is the name of an ITEM whose contents are to be compared with the comparand.

version-b is an unsigned integer in the range 1 to 15.

EQ (or =) means equal to.

NE means not equal to.

GT (or >) means greater than.

GE means greater than or equal to.

LT (or <) means less than or equal to.

literal is a literal comparand.

item-c is the name of an ITEM whose contents are the comparand.

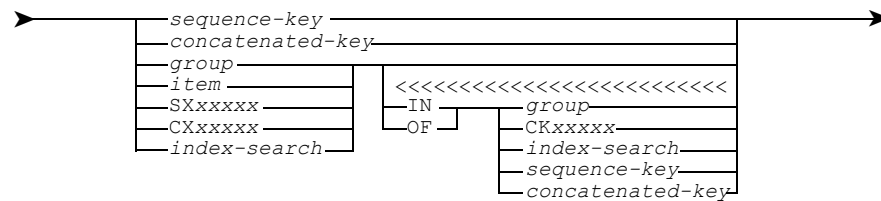
version-c is an unsigned integer in the range 1 to 15.

content is as defined above.

freq is an unsigned integer in the range 1 to 16777215.

sequence-key is a 1 to 8 character unique alphanumeric name.

generates-clause is:



where:

sequence-key is as described above.

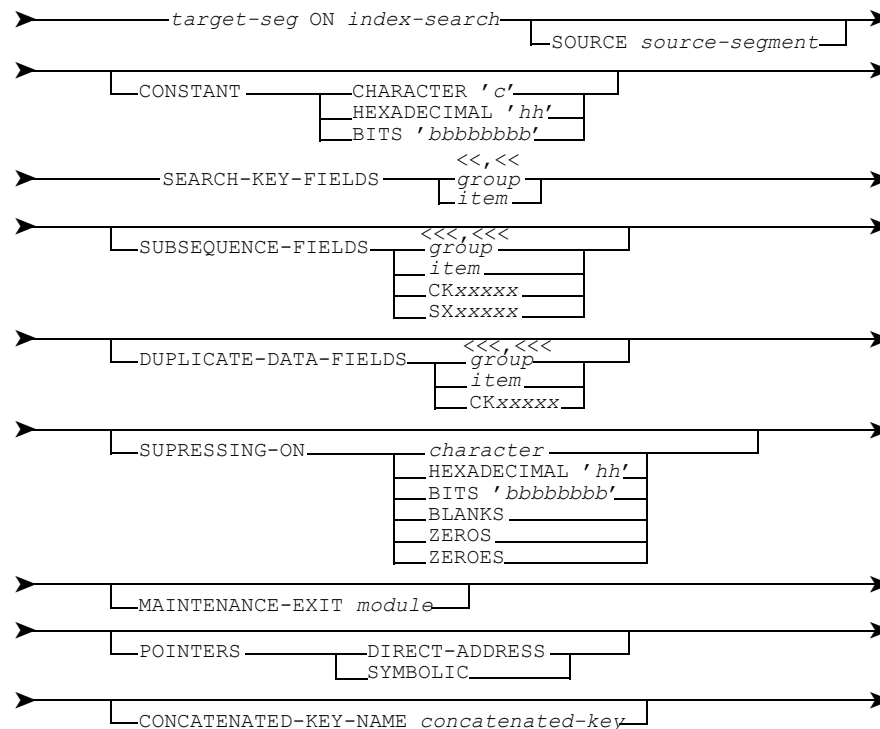
concatenated-key is used when a logical child segment is being defined, to specify the name to be given to the destination parent's concatenated key.

group and *item* are as defined above.

CKxxxxxx is a three- to seven-character name to be applied to system-related field within a GENERATES or SUBSEQUENCE-FIELDS clause. The name must be unique and must start with CK.

index-search is a 1 to 8 character unique alphanumeric name.

related-details are:



where:

target-seg is the name of a SEGMENT that is a PHYSICAL-TARGET-SEGMENT member.

index-search is a 1 to 8 character unique alphanumeric name.

source-segment is the name of a SEGMENT that is a PHYSICAL-SOURCE-SEGMENT member.

module is the name of a MODULE member.

c is any printable character.

hh is a hexadecimal representation of any printable or non-printable character.

bbbbbbbb is a one-byte bit string representation of any printable or non-printable character.

group and *item* are as defined above.

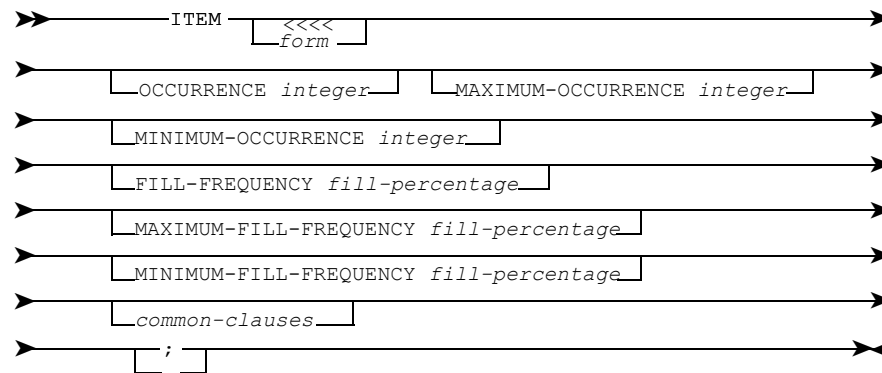
common-clauses are any of the clauses common to all member types.

INFOBANK-PANEL

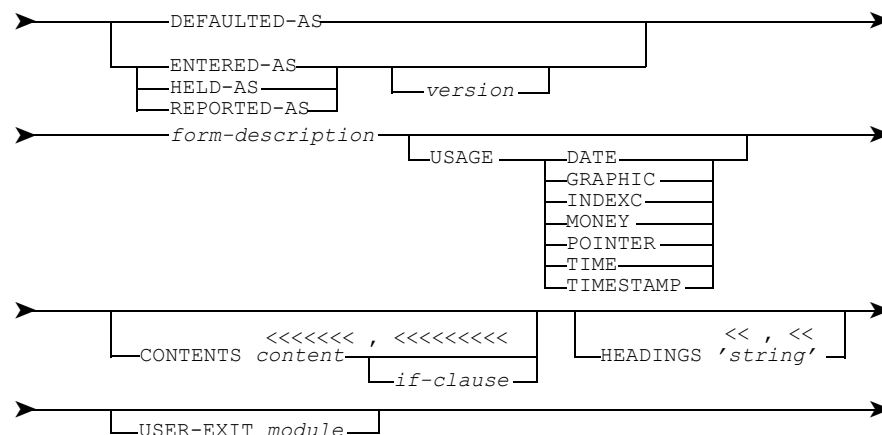
See *ASG-Manager Products Quick Reference: Administration* for further details of the INFOBANK-PANEL member type.

ITEM

To define a data element.



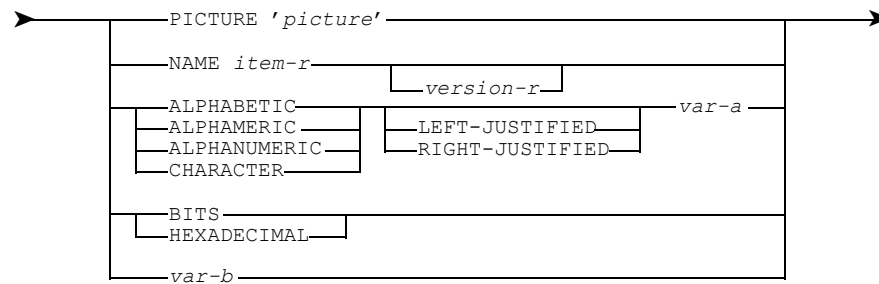
where *form* is:



where:

version is an unsigned integer in the range of 1 to 15.

form-description is:



where:

picture is a symbol or a string of not more than 30 characters. Picture symbols are:

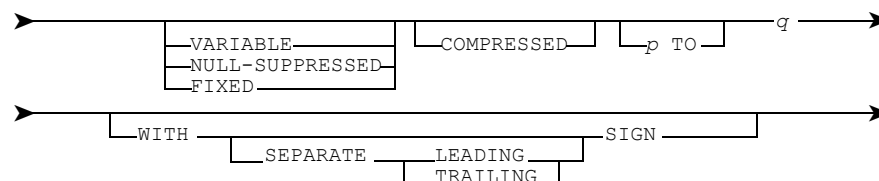
Picture Type	Format Symbol:																		
	A	B	CR	DB	E	I	K	P	R	S	T	V	X	Y	Z	0	9	£	+
ALPHABETIC	*																		
ALPHANUMERIC	*												*			*			
ALPHANUMERIC EDITED	*	*											*			*	*		
NUMERIC								*				*				*			
NUMERIC EDITED		*	*	*		*		*	*	*	*	*	*	*	*	*	*	*	*
NUMERIC FLOATING POINT					*	*	*		*		*	*				*		*	*

* indicates permissible combinations of symbols within picture types.

item-r is the name of an item.

version-r is an unsigned integer in the range 1 to 15.

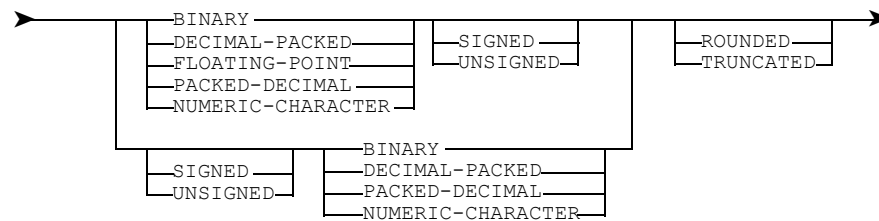
var-a is:



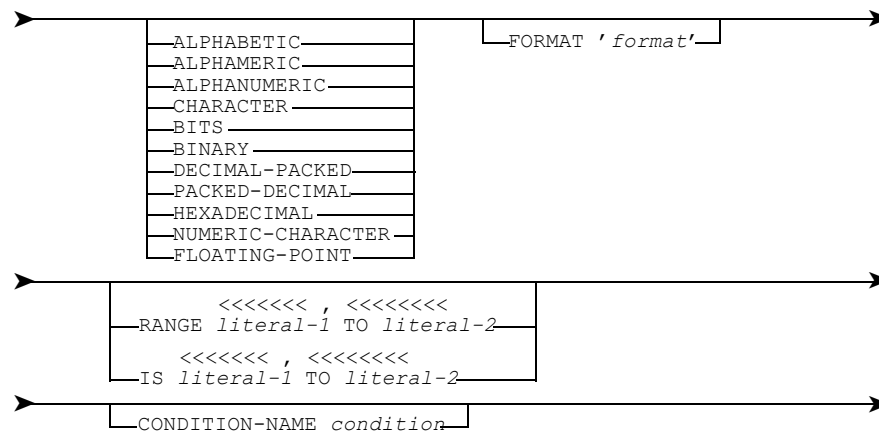
where:

p and *q* are unsigned decimal numbers specifying respectively the minimum and maximum lengths of the item.

var-b is:



content is:



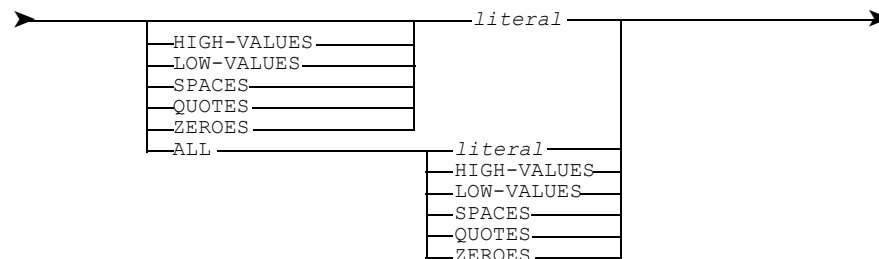
where:

'*format*' is a symbol or string of not more than 30 characters. Format symbols are:

Form-description	Format Symbol:										
	A	N	O	P	S	X	+	-	.	/	DMY
ALPHABETIC	*				*						
ALPHAMERIC	*	*	*	*	*		*	*	*		*
ALPHANUMERIC	*	*	*	*	*		*	*	*		*
BINARY							*	*	*		*
BITS											
CHARACTER	*	*			*	*			*	*	*
DECIMAL-PACKED		*					*	*	*		*
PACKED-DECIMAL		*					*	*	*		*
FLOATING-POINT		*					*	*	*		
HEXADECIMAL											
NUMERIC-CHARACTER		*	*	*			*	*	*		*
PICTURE	*	*	*	*	*	*	*	*	*	*	*

* indicates permissible combinations of symbols and keywords.

literal-1 and *literal-2* are:



where:

literal is:

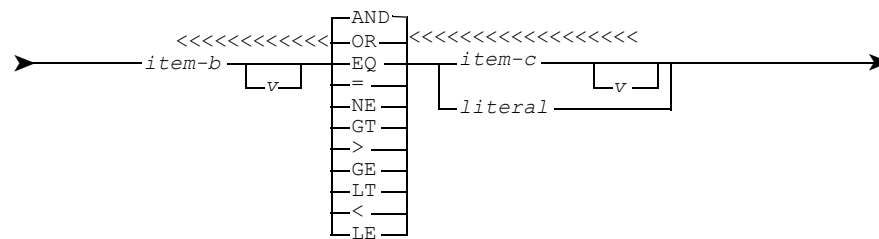
- A delimited character string of not more than 256 characters or a numeral in the range 1 to 9
- An undelimited signed or unsigned decimal number of not more than 18 digits, optionally with a decimal point
- An undelimited signed or unsigned floating point number

condition is a name of not more than 32 characters, conforming to the rules for member names.

if-clause is:



where *clause* is:



where:

item-b is the name of an ITEM to be compared with *item-c* or a literal.

v is an unsigned integer in the range 1 to 15.

item-c is the name of an ITEM to be compared with *item-b*.

EQ (or =) means equal to.

NE means not equal to.

GT (or >) means greater than.

GE means greater than or equal to.

LE means less than or equal to.

literal is the literal being compared with item-b.

content is as defined above.

'*string*' is a character string of not more than 256 printable characters. Spaces (hexadecimal 40) are regarded as printable characters.

module is the name of a module in which the item is processed.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

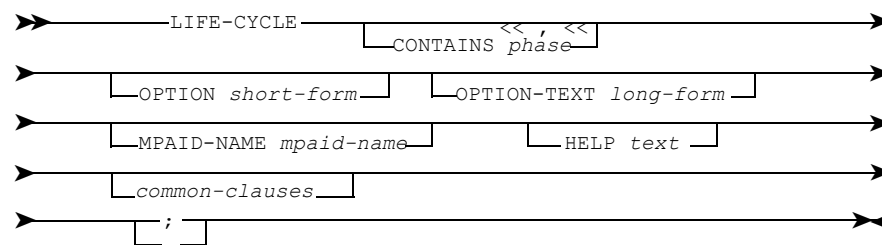
fill-percentage is an integer in the range 1 to 100.

text is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

common-clauses are any of the clauses common to all member types.

LIFE-CYCLE

To provide the methodological framework for application development.



where:

phase is the name of a PHASE member.

short-form is a delimited string of up to 32 characters.

long-form is a delimited string of up to 50 characters.

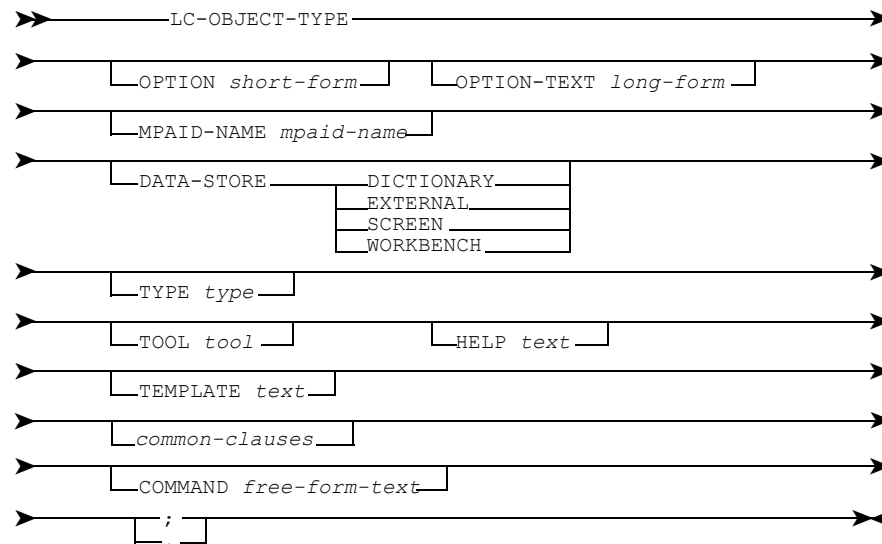
mpaid-name is a name of up to 10 characters.

text is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

common-clauses are any of the clauses common to all member types.

LIFE-CYCLE-OBJECT-TYPE

A formal description of a result.



where:

short-form is a delimited string of up to 39 characters.

long-form is a delimited string of up to 50 characters.

mpaid-name is a name of up to 10 characters.

type is O for optional or M for mandatory.

tool is a delimited string of up to 15 characters.

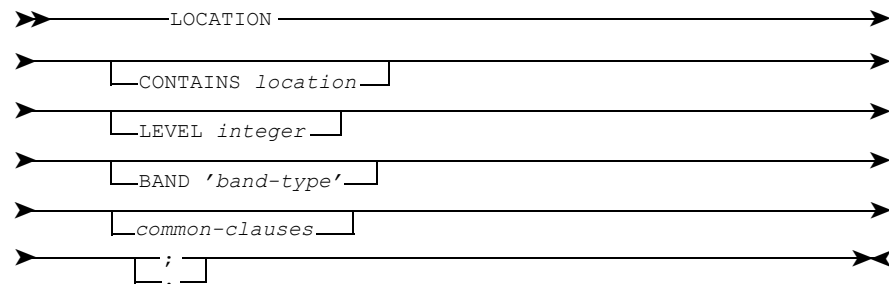
text is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

common-clauses are any of the clauses common to all member types.

free-form-text is a maximum of 32767 lines of text, each line containing a maximum of 248 characters.

LOCATION

The geographical place or area where an organizational unit exists.



where:

location is the name of a LOCATION member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

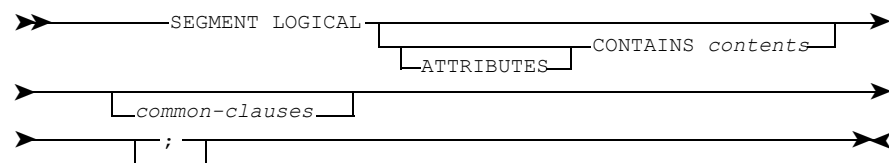
Note:

The delimiters shown in the above syntax are required when defining a LOCATION member via the command interface.

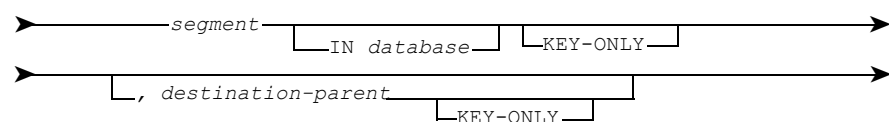
common-clauses are any of the clauses common to all member types.

LOGICAL-SEGMENT

A LOGICAL SEGMENT contains a physical segment or the concatenation of two physical segments.



where *contents* are:



where:

segment is the name of a PHYSICAL SEGMENT.

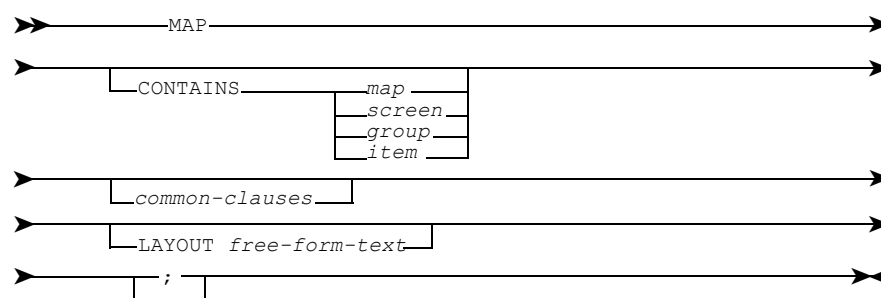
database is the name of a HISAM, HDAM, or HIDAM database.

destination-parent is a PHYSICAL DESTINATION-PARENT-SEGMENT.

common-clauses are any of the clauses available to all member types.

MAP

A set of data elements and literals comprising a logical screen image.



where:

map is the name of a MAP member.

screen is the name of a SCREEN member.

group is the name of a GROUP member.

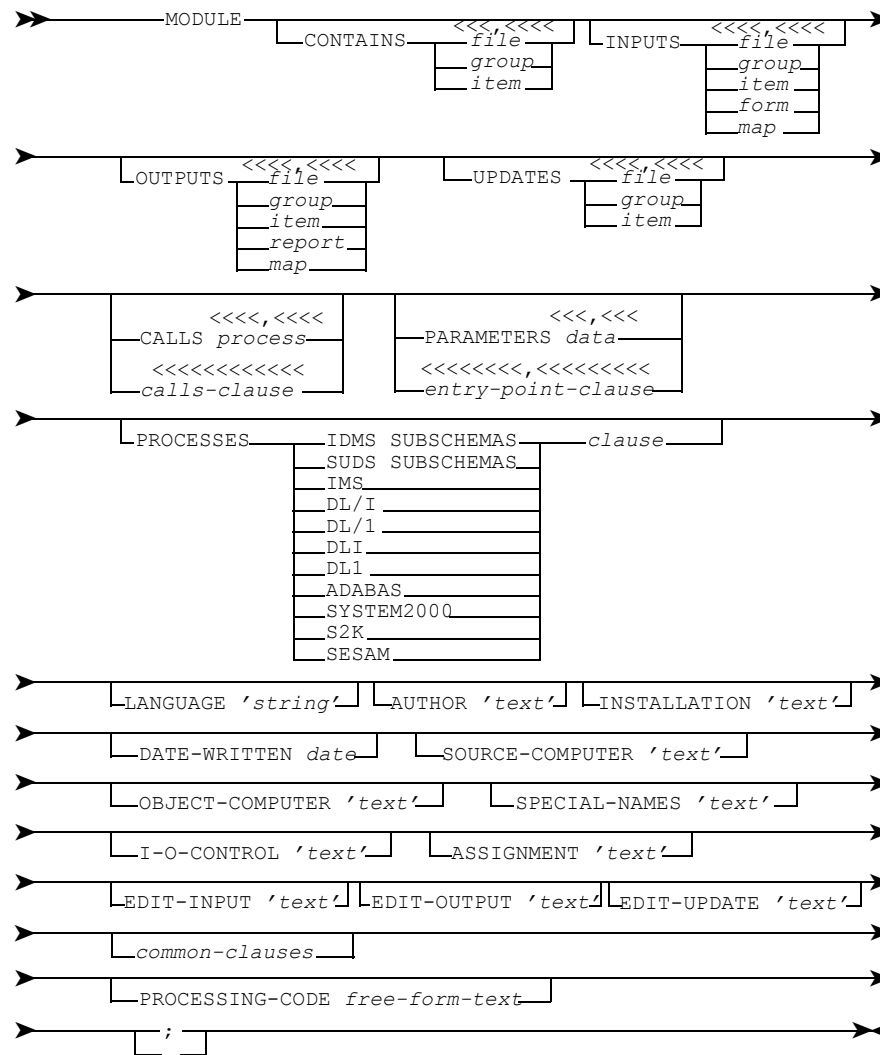
item is the name of an ITEM member.

common-clauses are any of the clauses common to all member types.

free-form-text is a maximum of 32767 lines of text, each containing a maximum of 248 characters.

MODULE

A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading, but cannot be executed on its own.



where:

file is the name of a FILE member.

group is the name of a GROUP member.

item is the name of an ITEM member.

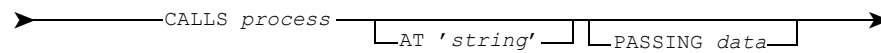
form is the name of a FORM member.

map is the name of a MAP member.

report is the name of a REPORT member.

process identifies a process member at the same or a lower level in the member type hierarchy.

calls-clause is:



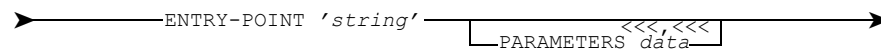
where:

process is as defined above.

'*string*' is a character string of not more than 256 characters.

data identifies a data member.

entry-point-clause is:



where:

string and *data* are as defined above.

clause is defined in the User Guide specific to your environment.

'*string*' is as defined above.

'*text*' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters. The delimiters mentioned above are required when defining a MODULE member via the command interface.

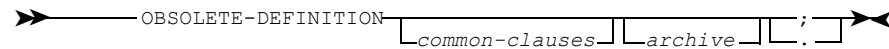
date is a date, in the format defined by your installation.

common-clauses are any of the clauses common to all member types.

free-form-text is a maximum of 32767 lines of text, each line containing a maximum of 248 characters.

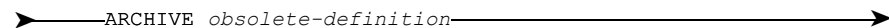
OBSOLETE-DEFINITION

To allow the retention of obsolete members within the repository, but in a form in which they are invisible to the current status and its dependent statuses.



common-clauses are any of the clauses common to all member types.

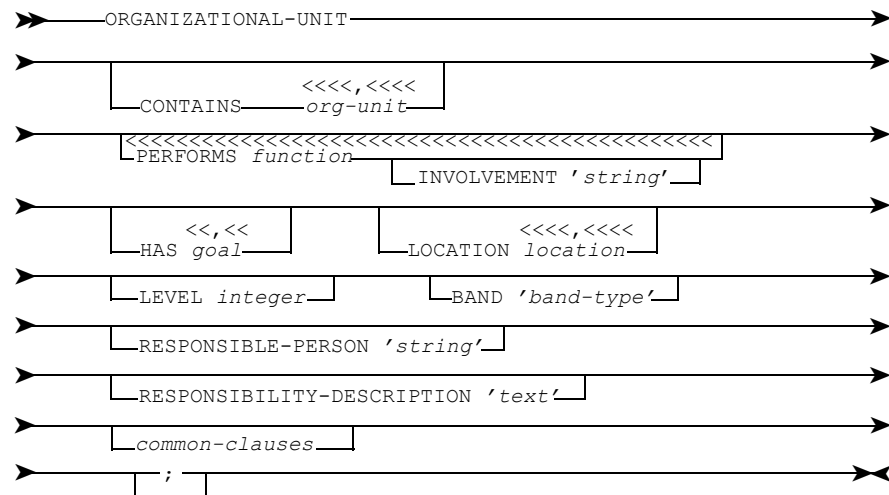
archive is:



where *obsolete-definition* is the full definition of the obsolete member.

ORGANIZATIONAL-UNIT

The organization or part thereof which can be considered as a self-contained unit for operations, duties, responsibilities, and authorities.



where:

org-unit is the name of an ORGANIZATIONAL-UNIT member.

function is the name of a FUNCTION member.

goal is the name of a GOAL member.

location is the name of a LOCATION member.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

'*string*' is a character string of not more than 256 characters.

'*text*' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

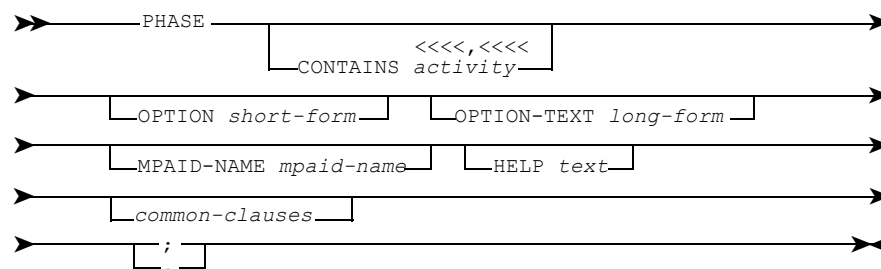
Note:

The delimiters shown in the above syntax are required when defining an ORGANIZATIONAL-UNIT member via the command interface.

common-clauses are any of the clauses common to all member types.

PHASE

To give the life cycle a structure.



where:

activity is the name of an ACTIVITY member.

short-form is a delimited string of up to 32 characters.

long-form is a delimited string of up to 50 characters.

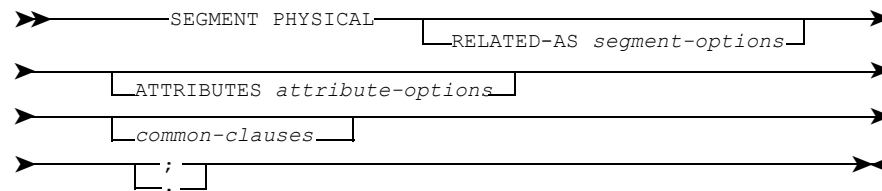
mpaid-name is a name of up to 10 characters.

text is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

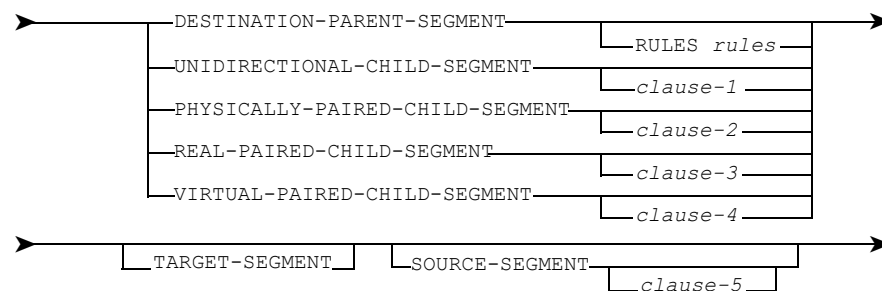
common-clauses are any of the clauses common to all member types.

PHYSICAL-SEGMENT

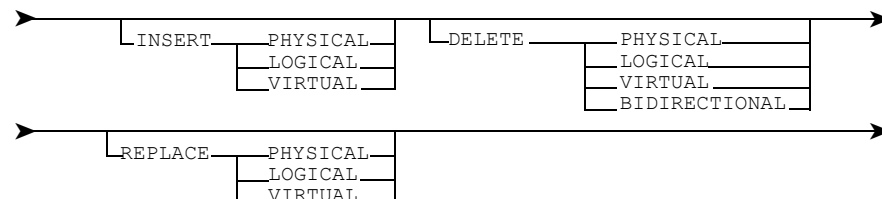
To define the structure of a database hierarchy.



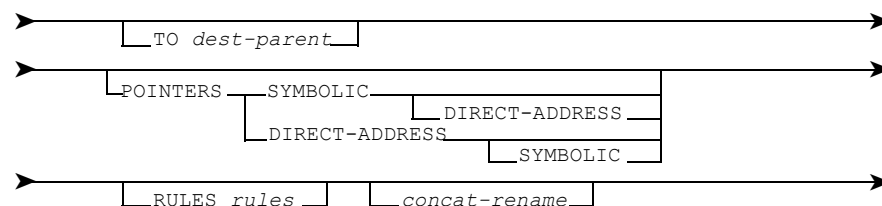
where *segment-options* are:



where *rules* are:



clause-1 is:

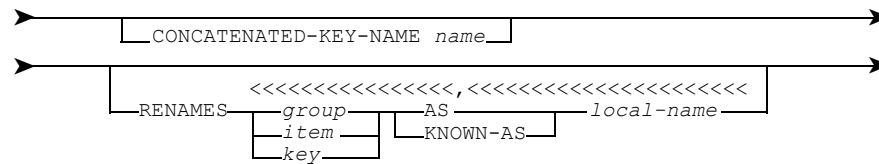


where:

dest-parent is the name of a PHYSICAL-DESTINATION-PARENT-SEGMENT member.

rules are as defined above.

concat-rename is:



where:

name is the name of a CONACATENATED-KEY member.

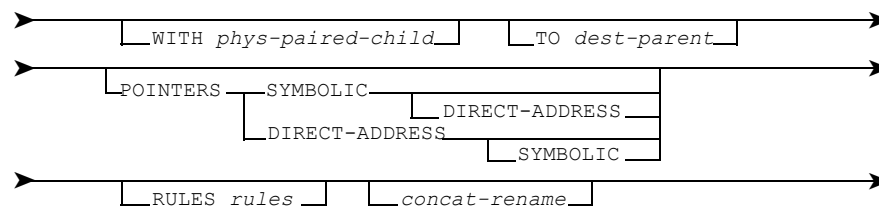
group is the name of a GROUP member.

item is the name of an ITEM member.

key is a 1 to 8 character unique alphanumeric name.

local-name is a name conforming to the rules for member names.

clause-2 is:

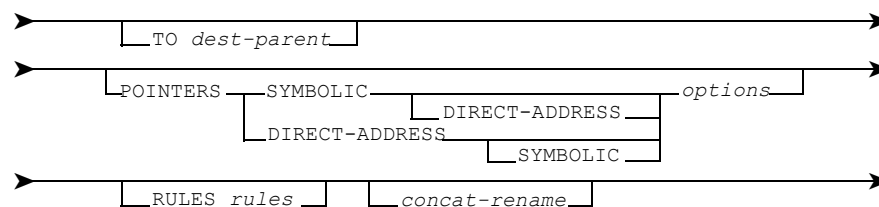


where:

phys-paired-child is the name of a PHYSICALLY-PAIRED-CHILD-SEGMENT member.

des-parent, *rules*, and *concat-rename* are as defined above.

clause-3 is:



where:

dest-parent, *rules*, and *concat-rename* are as defined above.

FORWARD-LOGICAL-TWIN SINGLE-LOGICAL-CHILD

BACKWARD-LOGICAL-TWIN DOUBLE-LOGICAL-CHILD

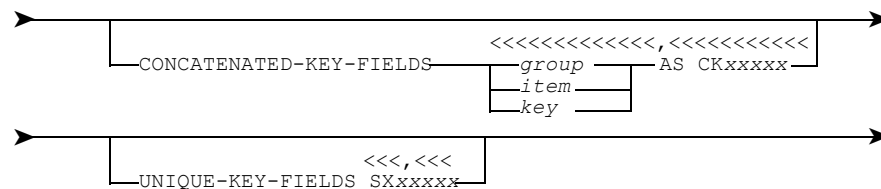
```

graph LR
    subgraph Row1 [ ]
        direction LR
        A[WITH real-paired-child] --- B[TO dest-parent]
    end
    subgraph Row2 [ ]
        direction LR
        C[concat-rename]
    end
    Row1 --> Row2

```

real-paired-child is the name of a PHYSICAL REAL-PAIRED-CHILD-SEGMENT member.

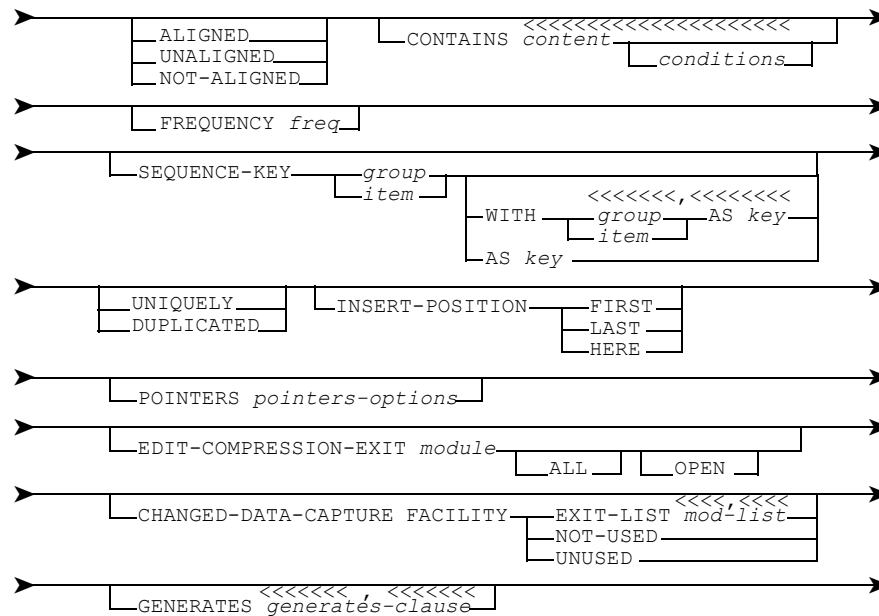
clause-5 is:



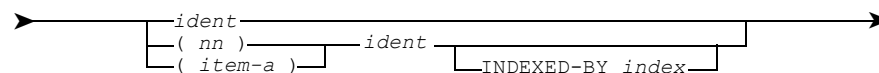
CKxxxxxx is a three- to seven-character name to be applied to a system related field within a GENERATES, SUBSEQUENCE-FIELDS, or DUPLICATE-DATA-FIELDS clause. The name must be unique and must start with CK.

SKxxxxx is a three- to seven-character name to be applied to a system related field within a GENERATES or SUBSEQUENCE-FIELDS clause. The name must be unique and must start with SX.

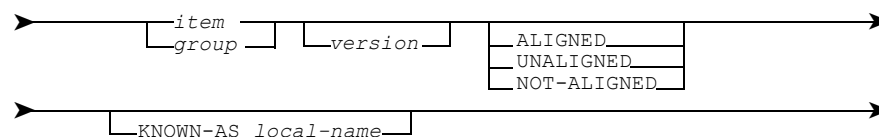
attribute-options are:



where *content* is:



where *ident* is:



where:

item and *group* are as defined above.

version is an unsigned integer in the range 1 to 15.

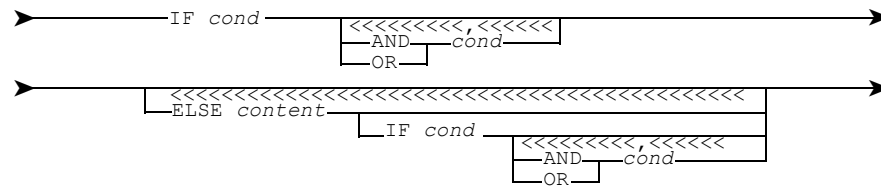
local-name is as defined above.

nn is an unsigned integer of from 1 to 18 digits, being the number of times *item* or *group* occurs in the array.

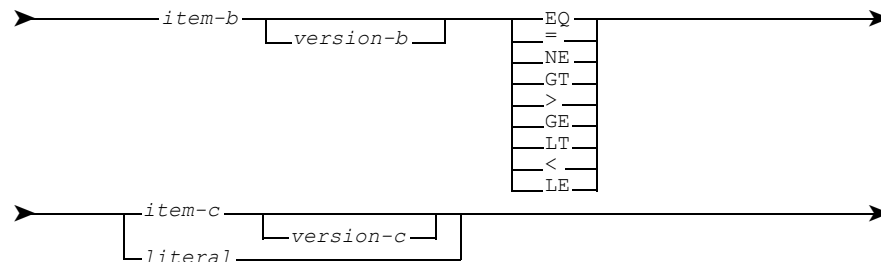
item-a is the name of an ITEM member.

index is a name, conforming to the rules for member names.

conditions are:



cond is:



where:

item-b is the name of an ITEM whose contents are to be compared with the comparand.

version-b is an unsigned integer in the range 1 to 15.

EQ (or =) means equal to.

NE means not equal to.

GT (or >) means greater than.

GE means greater than or equal to.

LT means less than or equal to.

LE means less than or equal to.

literal is a literal comparand.

item-c is the name of an ITEM whose contents are the comparand.

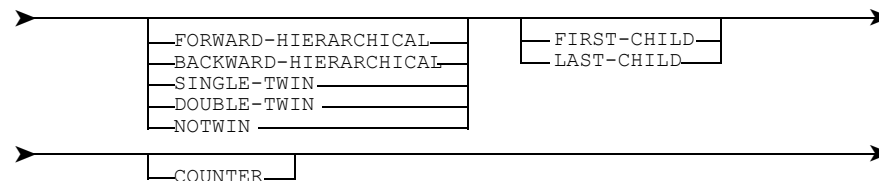
version-c is an unsigned integer in the range 1 to 15.

content is as defined above.

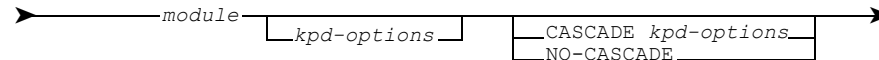
freq is an unsigned number in the range 0.01 to 16777215.00, or (for root segments) an integer in the range 1 to 16777215.

group, *item*, and *key* are as defined above.

pointers-options are:



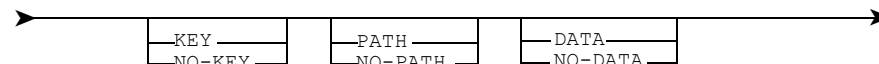
mod-list is:



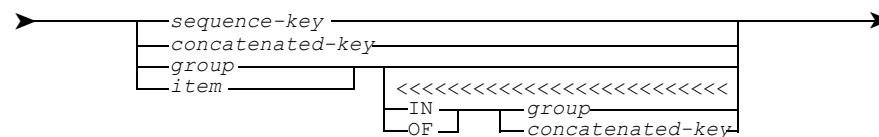
where:

module is the name of a MODULE or PROGRAM member. Dummies are created as modules.

kpd-options are:



generates-clause is:



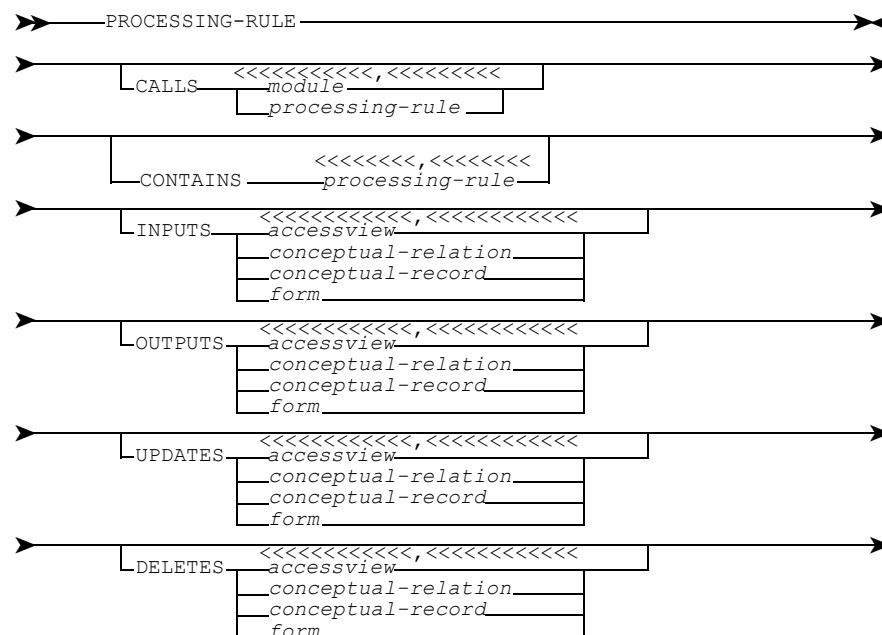
where:

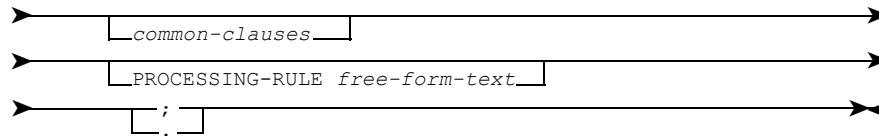
sequence-key is a 1 to 8 character unique alphanumeric name.

concatenated-key specifies the name to be given to the destination parent's concatenated key.

group and *item* are as defined above.

common-clauses are any of the clauses common to all member types.





where:

module is the name of a MODULE member.

processing-rule is the name of a PROCESSING-RULE member.

accessview is the name of an ACCESSVIEW member.

conceptual-relation is the name of a CONCEPTUAL-RELATION member.

form is the name of a FORM member.

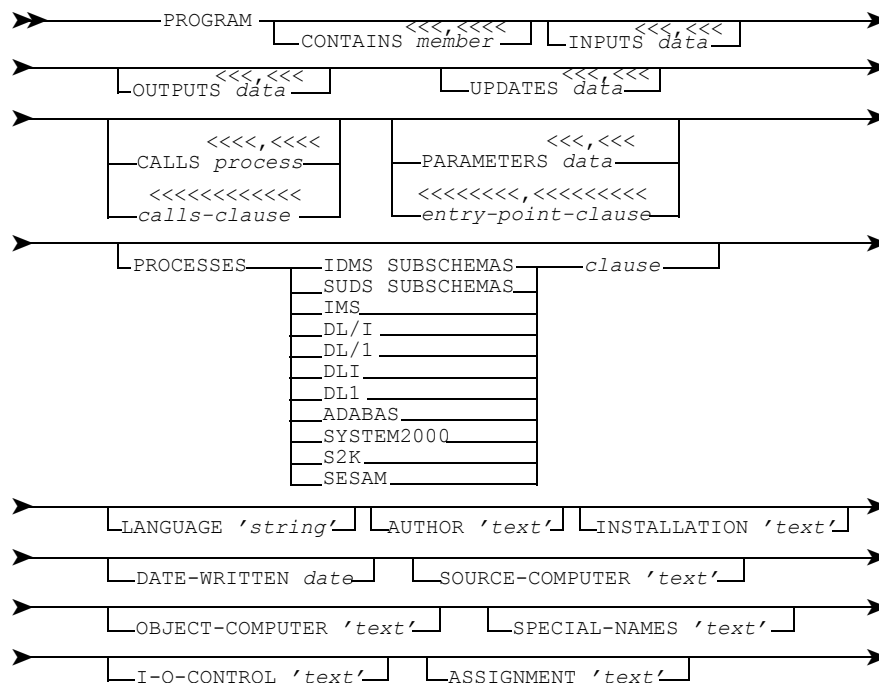
conceptual-record is the name of a CONCEPTUAL-RECORD member.

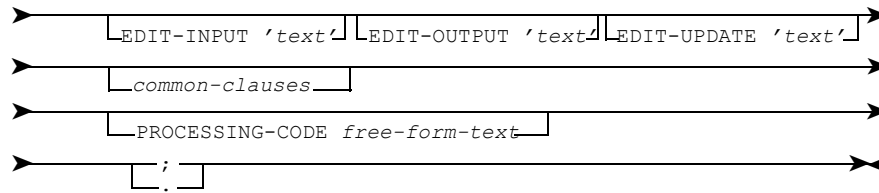
common-clauses are any of the clauses common to all member types.

free-form-text is a maximum of 32767 delimited character strings, each containing a maximum of 248 characters.

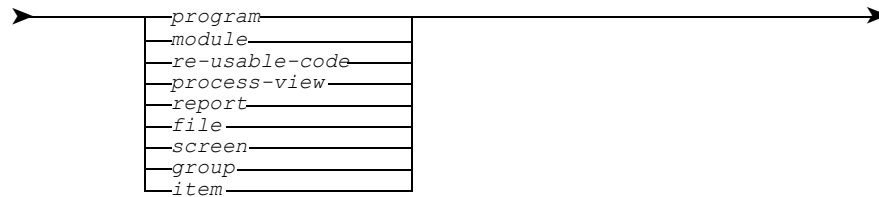
PROGRAM

A set of actions or instructions that a machine is capable of executing as a whole.

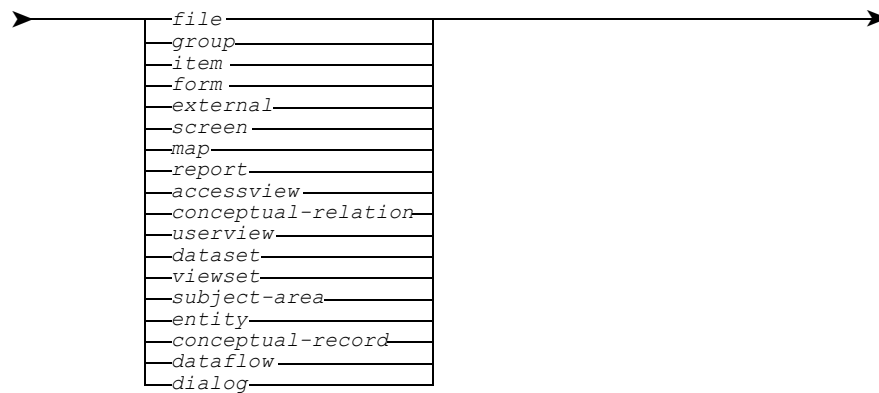




where *member* is:

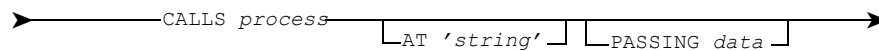


data is:



process identifies a process member at the same or a lower level in the member type hierarchy.

class-clause is:



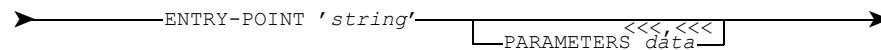
where:

process is as defined above.

'*string*' is a character string of not more than 256 characters.

data is as defined above.

entry-point-clause is:



where:

'string' and *data* are as defined above.

clause is defined in the User Guide specific to your environment.

'string' is as defined above.

'text' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

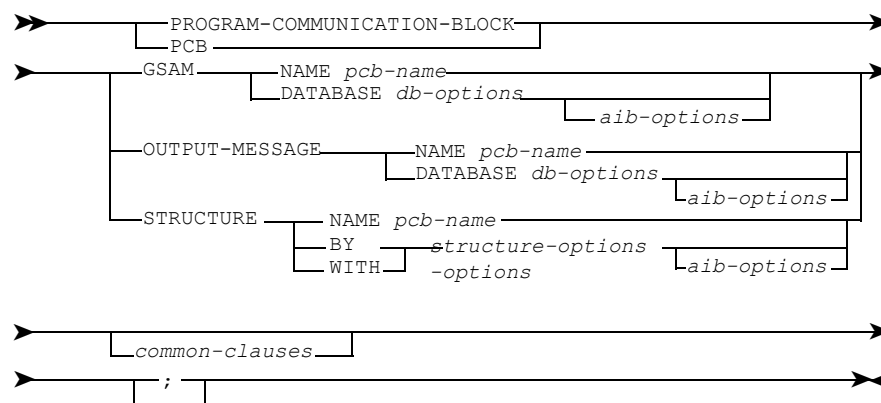
date is a date, in the format defined by your installation.

common-clauses are any of the clauses common to all member types.

free-form-text is maximum is a maximum of 32767 lines of text, each containing a maximum of 248 characters.

PROGRAM-COMMUNICATION-BLOCK

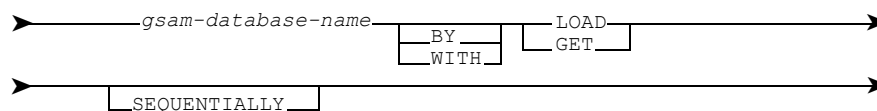
Defines program communication blocks to access IMS (DL/I) databases and message queues.



where:

pcb-name is the name of another PROGRAM-COMMUNICATION-BLOCK member.

db-options are:



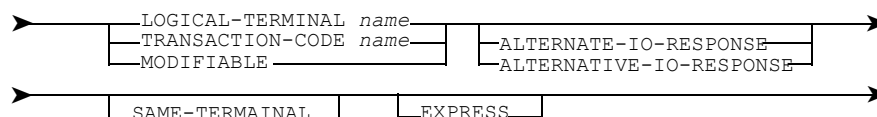
where:

gsam-database-name is the name of a database member of the GSAM type.

aib-options are:



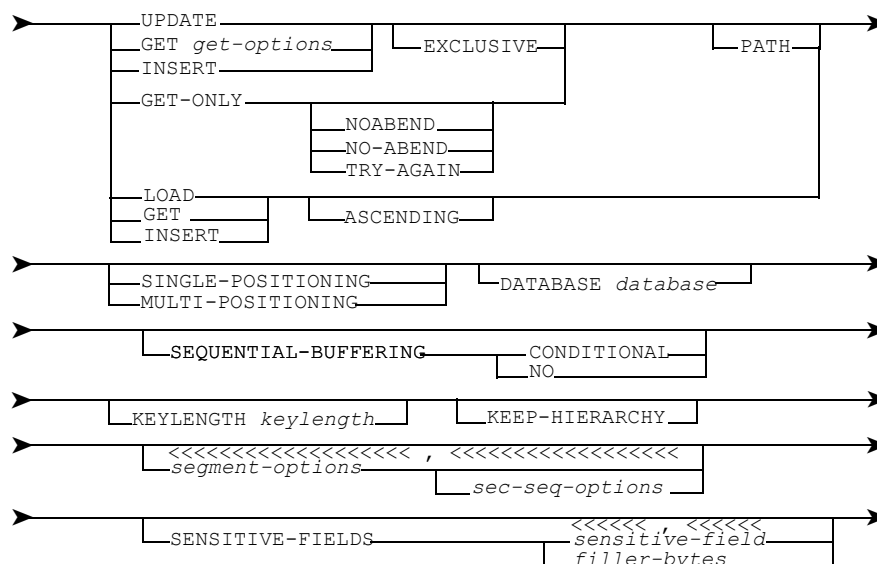
out-options are:



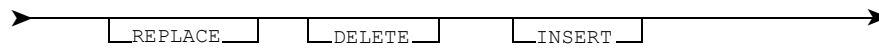
where:

name is an alphanumeric name 1 to 8 characters in length.

structure-options are:



where *get-options* are:



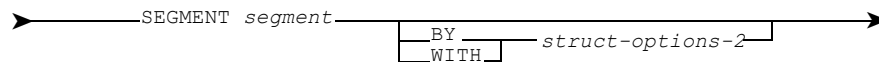
Note:

The UPDATE, GET, INSERT, LOAD, EXCLUSIVE, ASCENDING, PATH, GET ONLY, NOABEND, TRY-AGAIN, REPLACE, and DELETE keywords can all be optionally separated by commas.

database is the name of a DATABASE member.

keylength is an integer in the range 0 to 32767.

segment-options are:



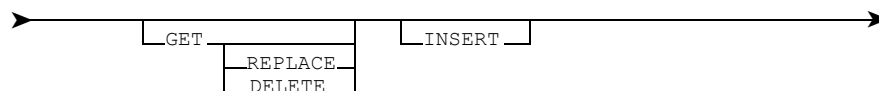
where:

segment is the name of a SEGMENT.

member struct-options-2 are:



where *grdi-options* are:



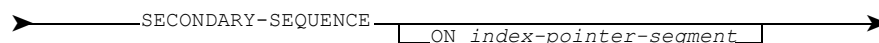
Note:

The UPDATE, KEY-SENSITIVE, GET REPLACE, GET DELETE, INSERT, EXCLUSIVE, and PATH keywords can all be optionally separated by commas.

Note:

You must specify at least one keyword in *grdi-options*.

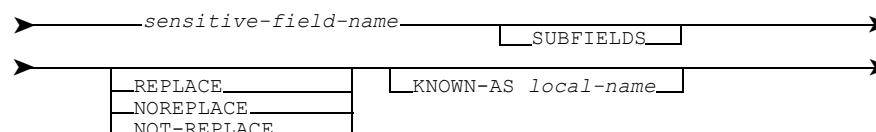
sec-seq-options are:



where:

index-pointer-segment is the name of an INDEX-POINTER SEGMENT member.

sensitive-field is:



where:

local-name is a name, conforming to the rules for member names.

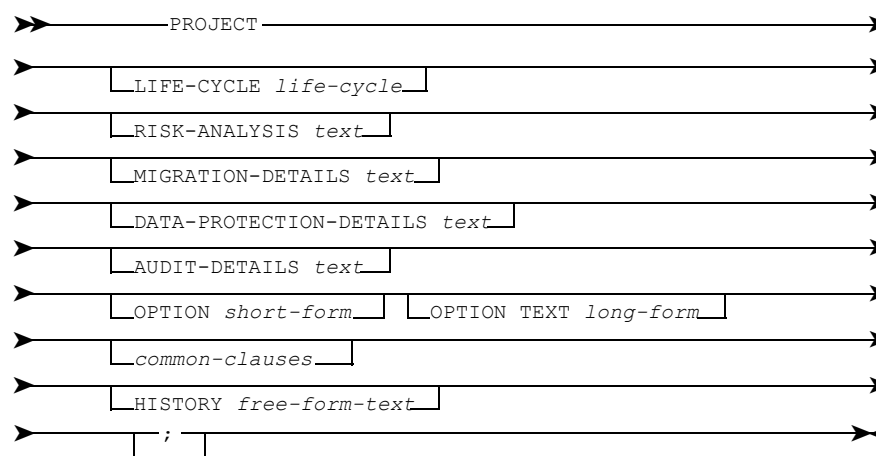
sensitive-field-name is the name of a GROUP, ITEM, or sequence key member or concatenated key member.

filler-bytes is an unsigned integer in the range 1 to 32767.

common-clauses are as defined in [Appendix B, "Common Clauses," on page 305](#).

PROJECT

The planning and allocation of resources for a period for the accomplishment of a predefined goal.



where:

life-cycle is the name of a LIFE-CYCLE member.

text is a maximum of 32767 delimited character strings, each containing a maximum of 256 characters.

short-form is a delimited string of up to 32 characters.

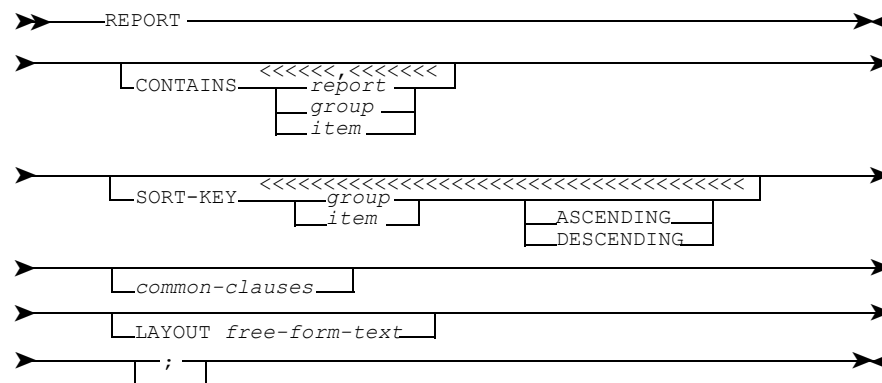
long-form is a delimited string of up to 50 characters.

common-clauses are any of the clauses common to all member types.

free-form-text is maximum of 32767 lines of text, each containing a maximum of 256 characters.

REPORT

A formal presentation of information, usually in the form of printed output from a computer.



where:

report is the name of a REPORT member.

group is the name of GROUP member.

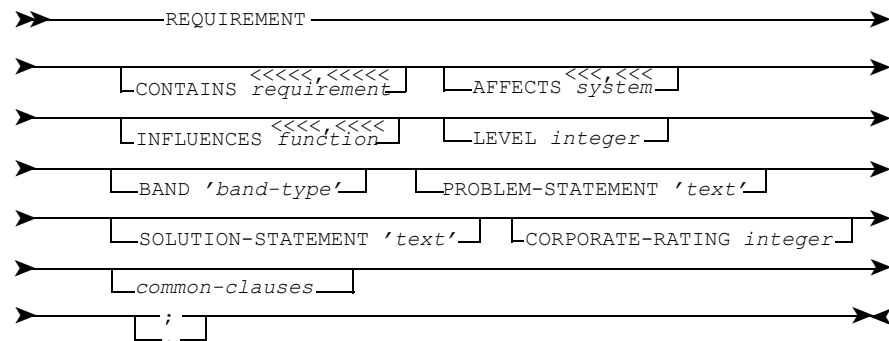
item is the name of an ITEM member.

common-clauses are any of the clauses common to all member types.

free-form-text is a maximum of 32767 lines of text, each containing a maximum of 256 characters.

REQUIREMENT

A statement of an informational need to solve a problem or exploit an opportunity for an organizational unit.



where:

requirement is the name of a REQUIREMENT member.

system is the name of a SYSTEM member.

function is the name of FUNCTION MEMBER.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

'*band-type*' is a text string of up to 78 characters, including delimiters.

'text' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

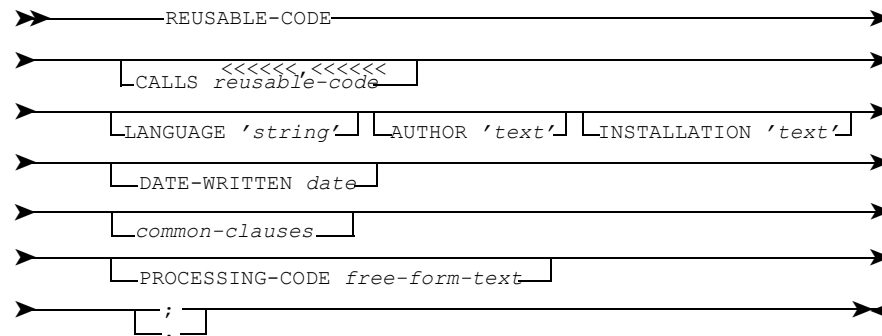
Note:

The delimiters shown in the above syntax are required when defining a REQUIREMENT member via the command interface.

common-clauses are any of the clauses common to all member types.

REUSABLE-CODE

A piece of program source code which can be incorporated in a module or program at compile time, in a form which may vary depending on the specified values of parameters in the inclusion clause.



where:

'reusable-code' is the name of a REUSABLE-CODE member.

'string' is a character string of not more than 256 characters.

'text' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

Note: _____

The delimiters are needed when defining a REUSABLE-CODE member via the command interface.

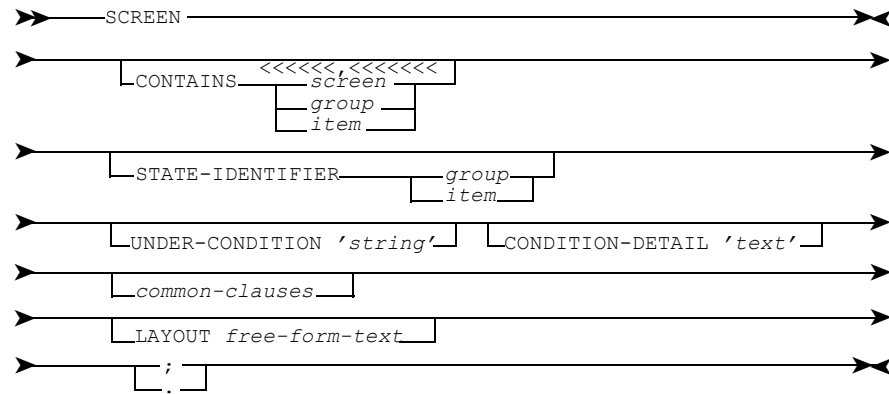
date is a date, in the format defined by your installation.

common-clauses are any of the clauses common to all member types.

free-form-text is a maximum of 32767 lines of text, each containing a maximum of 248 characters.

SCREEN

A set of data elements and literals normally comprising exactly one full screen on an interactive terminal display.



where:

screen is the name of a SCREEN member.

group is the name of a GROUP member.

item is the name of ITEM member.

'string' is a character string of not more than 256 characters.

'text' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

Note: _____

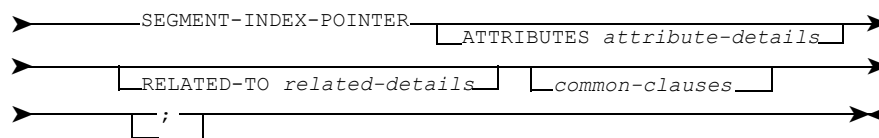
The delimiters shown in the above syntax are required when defining a SCREEN member via the command interface.

common-clauses are any of the clauses common to all member types.

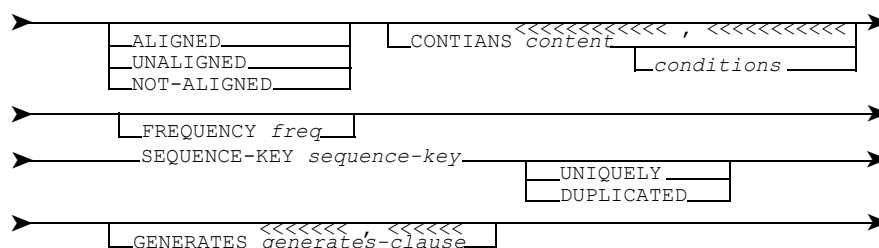
free-form-text is a maximum of 32767 lines of text, each containing a maximum of 248 characters.

SEGMENT INDEX-POINTER

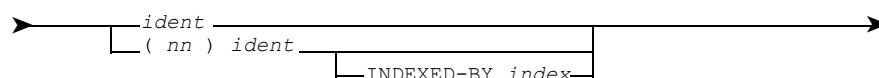
To define segments that reside in a secondary index database.



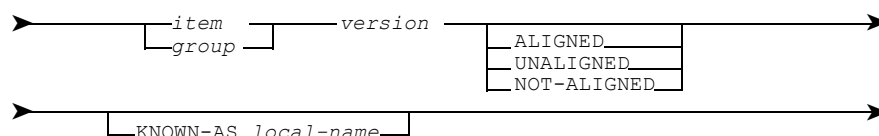
where *attribute-details* are:



where *content* is:



where *ident* is:



where:

item is the name of an ITEM repository member.

group is the name of a GROUP repository member.

version is an unsigned integer in the range 1 to 15.

local-name is a name, conforming to the rules for member names.

nn is an unsigned integer of from 1 to 18 digits, being the number of times item or group occurs in the array.

index is a name, conforming to the rules for member names, that is to be used as the index name when COBOL data descriptions are generated by the Source Language Generation facility.

Diagram illustrating the structure of an if-else statement in assembly language:

- The top line represents the **IF** branch, starting with `IF cond`. It contains a block of assembly code (AND, OR, etc.).
- The bottom line represents the **ELSE** branch, starting with `ELSE content`. It contains another block of assembly code.
- Arrows indicate the flow of execution from left to right.

The diagram illustrates the comparison of two items, *item-b* and *item-c*, based on their version and literal values. *item-b* is compared against *item-c* using a series of comparison operators: EQ, NE, GT, >, GE, LT, <, and LE. The diagram shows that *item-b* is greater than *item-c* (GT) and *item-b* is less than *item-c* (LT).

literal is a literal comparand.

version-b is an unsigned integer in the range 1 to 15.

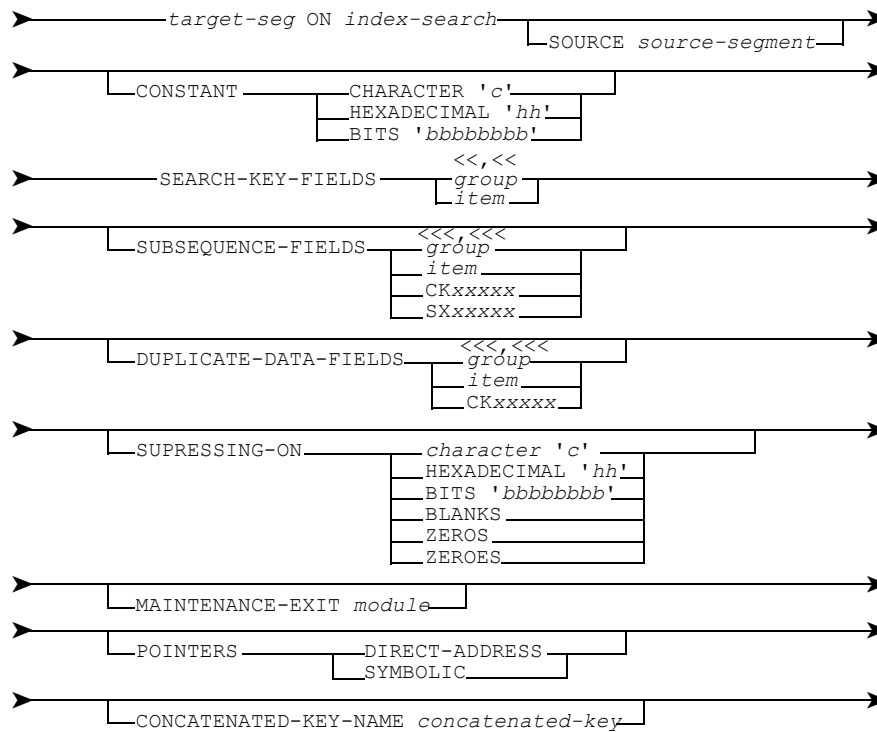
version-c is an unsigned integer in the range 1 to 15.

content is as defined above.

freq is an unsigned integer in the range 1 to 16777215.

sequence-key is a 1 to 8 character unique alphanumeric name.

related-details are:



where:

target-seg is the name of a SEGMENT that is a PHYSICAL TARGET-SEGMENT member.

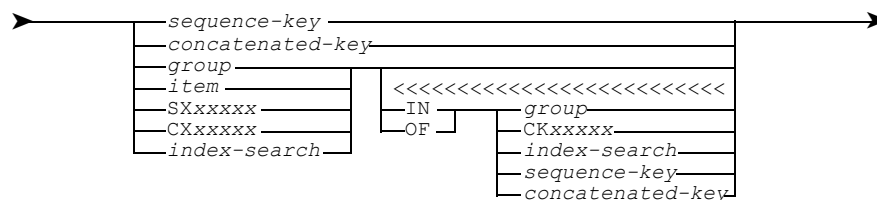
index-search-field is a 1 to 8 character unique alphanumeric name.

source-segment is the name of a SEGMENT that is a PHYSICAL SOURCE-SEGMENT member.

module is the name of a MODULE member.

group and *item* are as defined above.

generates-clause is:



where:

CKxxxxxx is a 3 to 7 character name to be applied to a system related field within a GENERATES, SUBSEQUENCE-FIELDS, or DUPLICATE-DATA-FIELDS clause. The name must be unique and start with CK.

SXxxxxxx is a 3 to 7 character name to be applied to a system related field within a GENERATES or SUBSEQUENCE-FIELDS clause. The name must be unique and start with SX.

concatenated-key is used when a logical child segment is being defined, to specify the name to be given to the destination parent's concatenated key.

c is any printable character.

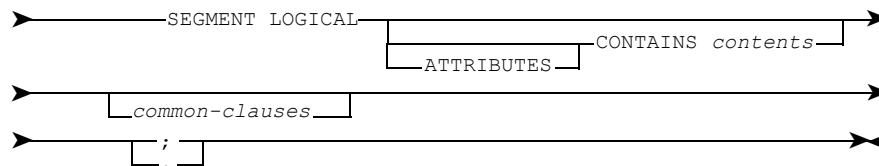
hh is a hexadecimal representation of any printable or non-printable character.

bbbbbbbb is a 1 byte bit string representation of any printable or non-printable character.

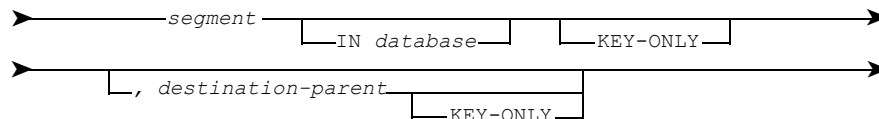
common-clauses are as defined in [Appendix B, "Common Clauses," on page 305](#).

SEGMENT LOGICAL

To define segments that reside in a logical database and to concatenate logically related physical segments.



where *contents* are:



where:

segment is the name of a PHYSICAL SEGMENT.

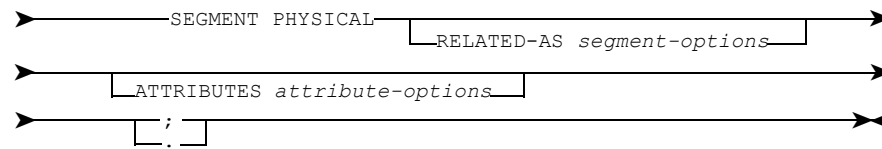
database is the name of a HISAM, HDAM, or HIDAM database.

destination-parent is a PHYSICAL DESTINATION-PARENT-SEGMENT.

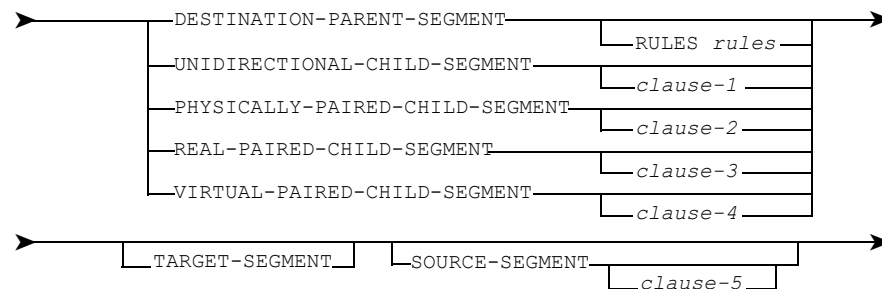
common-clauses are as defined in [Appendix B, "Common Clauses," on page 305](#).

SEGMENT PHYSICAL

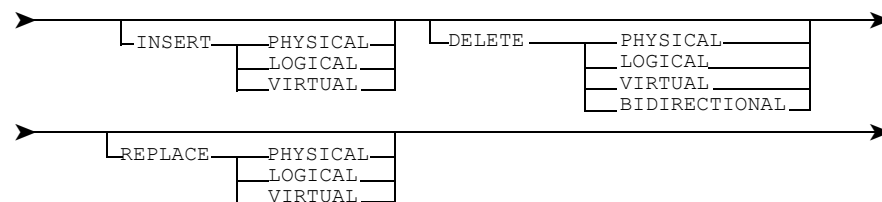
To define segments that reside in a physical database.



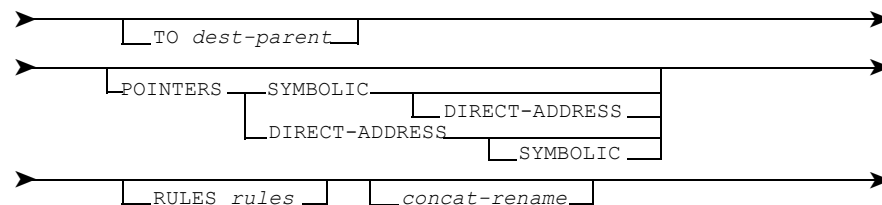
where *segment-options* are:



where *rules* are:



clause-1 is:



where:

dest-parent is the name of a PHYSICAL-DESIGNATION-PARENT-SEGMENT member.

```

    _CONCATENATED-KEY-NAME name
    RENAME <group> AS <local-name>
    <item> KNOWN-AS
    <key>
  
```

The diagram illustrates the structure of the 'rules' field in the 'rules' table. It is divided into two main sections: 'WITH phys-paired-child' and 'TO dest-parent'. The 'WITH' section is further divided into 'POINTERS' and 'SYMBOLIC'. The 'TO' section is further divided into 'DIRECT-ADDRESS' and 'SYMBOLIC'.

The diagram illustrates the structure of the 'to' field, which is a 32-bit field. It is divided into several sub-fields:

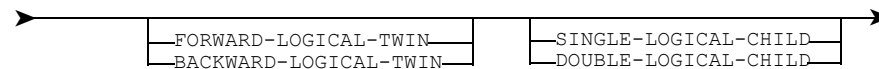
- TO dest-parent**: A 16-bit field at the top.
- POINTERS**: A 16-bit field below the first one.
- SYMBOLIC**: A 16-bit field below the second one.
- DIRECT-ADDRESS**: A 16-bit field below the third one.
- options**: A 16-bit field at the bottom right.
- RULES rules**: A 16-bit field at the bottom left.
- concat-rename**: A 16-bit field at the bottom right.

The fields are arranged in a hierarchical manner, with the top field being the most significant and the bottom field being the least significant.

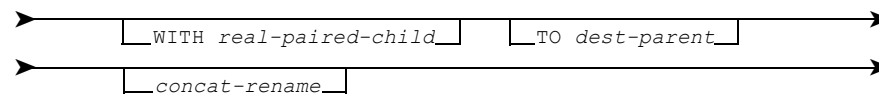
where:

dest-parent, *concat-rename*, *phys-paired-child*, and *rules* are as defined above.

options are:



clause-4 is:

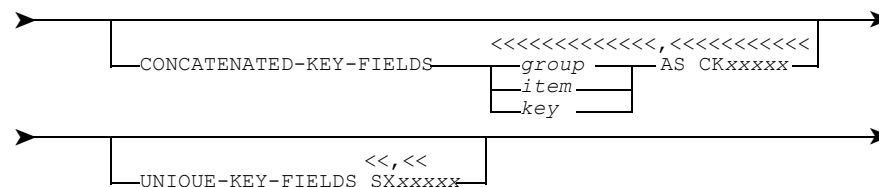


where:

real-paired-child is the name of a PHYSICAL REAL-PAIRED-CHILD-SEGMENT member.

des-parent and *concat-rename* are as defined above.

clause-5 is:



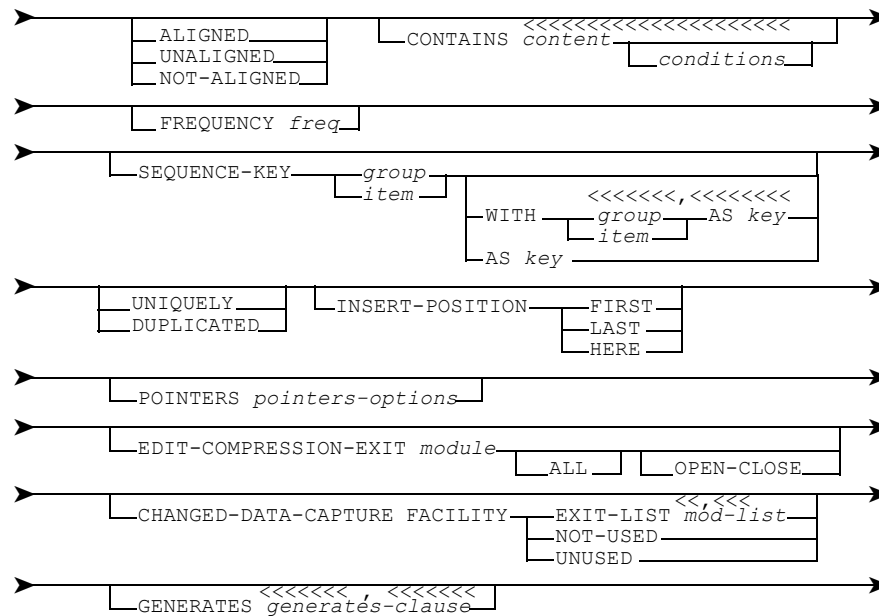
group, *item*, and *key* are as defined above.

CKxxxxxx is a 3- to 7-character name to be applied to a system related field within a GENERATES, SUBSEQUENCE-FIELDS, OR DUPLICATE-DATA-FIELDS clause. The name must be unique and must start with CK.

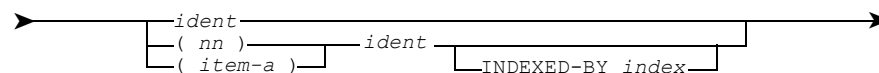
SKxxxxxx is a 3- to 7-character name to be applied to a system related field within a **GENERATES** or **SUBSEQUENCE-FIELDS** clause. The name must be unique and must start with **SX**.

key-name is a 1 to 8 character unique alphanumeric name.

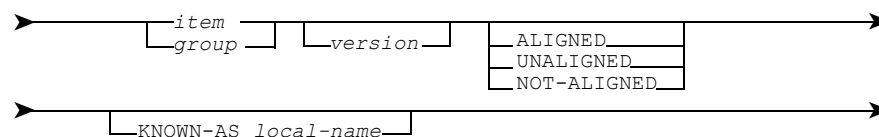
attribute-options are:



where *content* is:



where *ident* is:



where:

item and *group* are as defined above.

version is an unsigned integer in the range 1 to 15.

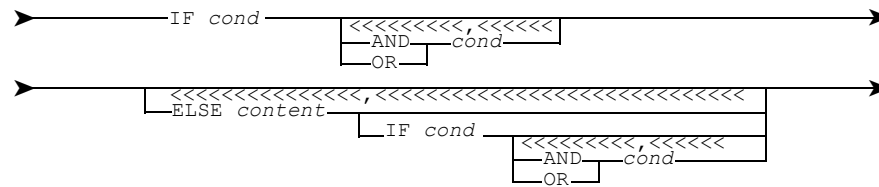
local-name is as defined above.

nn is an unsigned integer of from 1 to 18 digits, being the number of times *item* or *group* occurs in the array.

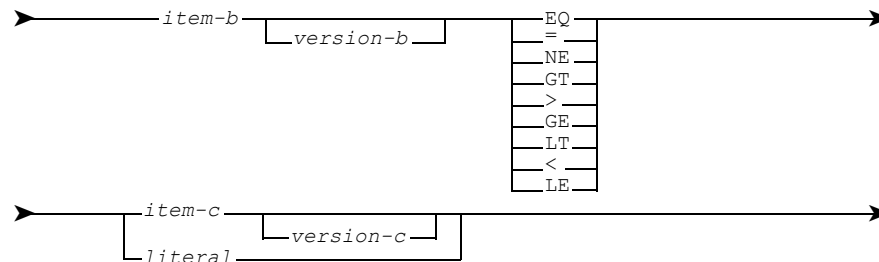
item-a is the name of an ITEM member.

index is a name, conforming to the rules for member names.

conditions are:



where *cond* is:



where:

literal is a literal comparand.

item-b is the name of an ITEM whose contents are to be compared with the comparand.

version-b is an unsigned integer in the range 1 to 15.

item-c is the name of an ITEM whose contents are the comparand.

version-c is an unsigned integer in the range 1 to 15.

freq is an unsigned number in the range 0.01 to 16777215.00, or (for root segments) an integer in the range 1 to 16777215.

content is as defined above.

name is the name of a CONCATENATED-KEY member.

module is the name of a MODULE member.

freq, *group*, *item*, *key*, and *module* are as defined above.

Diagram illustrating the data structure for the twinship test. The structure consists of two main horizontal timelines, each with a right-pointing arrow. The top timeline contains a box with five entries: FORWARD-HIERARCHICAL, BACKWARD-HIERARCHICAL, SINGLE-TWIN, DOUBLE-TWIN, and NOTWIN. The bottom timeline contains a box with one entry: COUNTER. To the right of the top timeline, there is a box with two entries: FIRST-CHILD and LAST-CHILD.

module kpd-options CASCADE kpd-options
NO-CASCADE

module is the name of a MODULE or PROGRAM member. Dummies are created as modules.

Diagram illustrating a timeline with three segments: KEY, PATH, and DATA. Each segment has a top bar and a bottom bar. The top bars are labeled KEY, PATH, and DATA. The bottom bars are labeled NO-KEY, NO-PATH, and NO-DATA.

[illegible]

concatenated-key specifies the name to be given to the destination parent's concatenated key.

sequence-key is a 1 to 8 character unique alphanumeric name.

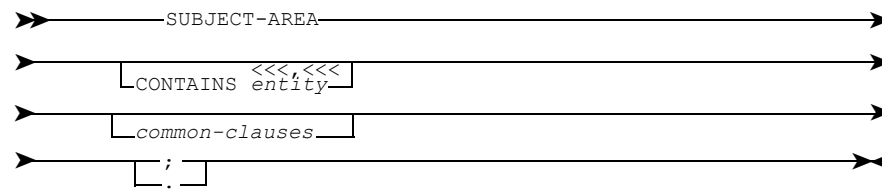
common-clauses are as defined in [Appendix B, "Common Clauses,"](#) on page 305.

SEXEC

See [Chapter 3, "Procedures Language," on page 295](#) for further details of the SEXEC member type.

SUBJECT-AREA

A grouping of entities which are logically related because of their involvement in similar business functions.



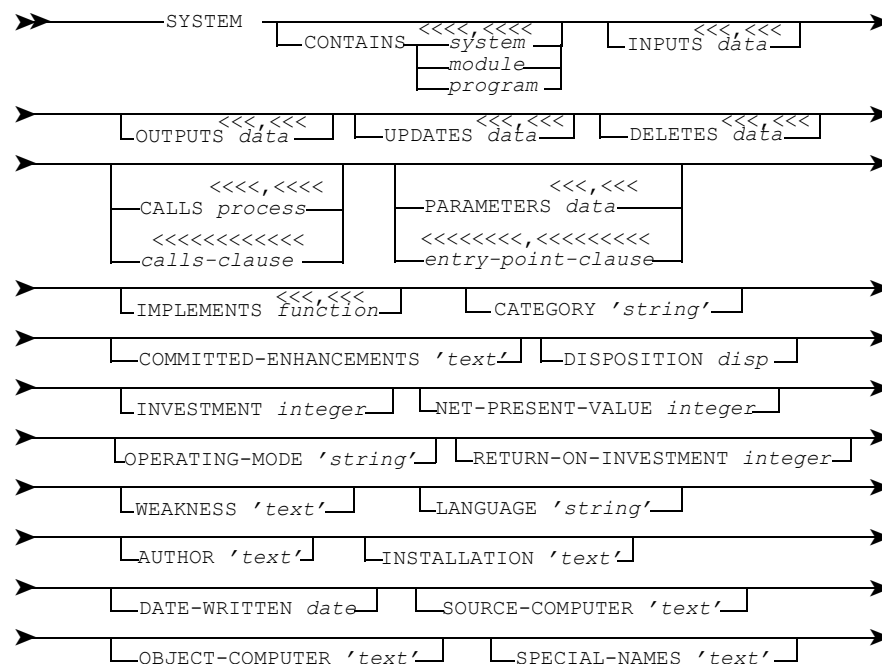
where:

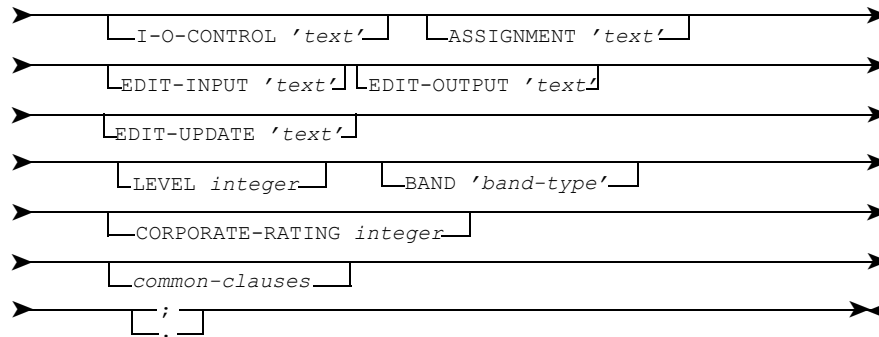
entity is the name of an ENTITY member.

common-clauses are any of the clauses common to all member types.

SYSTEM

A set of manual and automated procedures and associated documentation which work together to satisfy one or more informational needs of the organization.





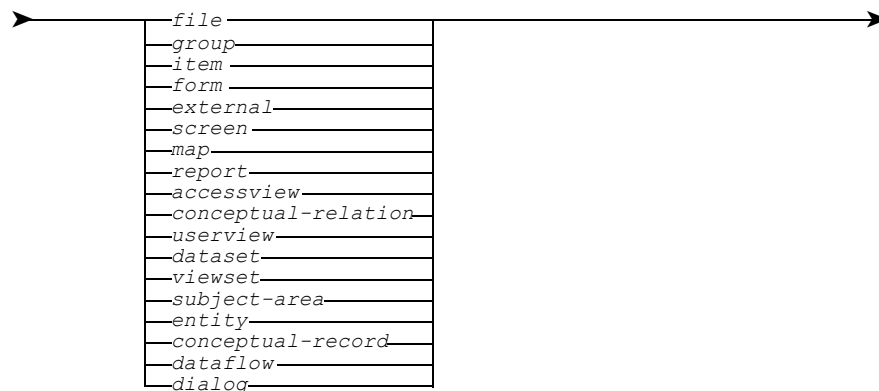
where:

system is the name of a SYSTEM member.

module is the name of a MODULE member.

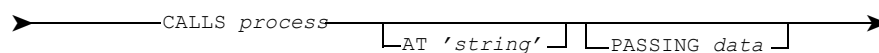
program is the name of a PROGRAM member.

data is the name of a member of any of these types:



process identifies a process member at the same or a lower level in the member type hierarchy.

calls-clause is:



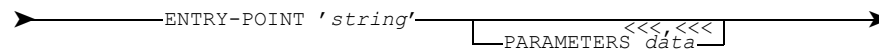
where:

process is as defined above.

'*string*' is a character string of not more than 256 characters.

data identifies a data member.

entry-point-clause is:



where:

string and *data* are as defined above.

function is the name of a FUNCTION member.

'*string*' is a defined above.

'*text*' is a maximum of 32767 delimited character strings, each containing a maximum of 246 characters.

disp is a string continuing a maximum of 78 characters.

integer is an integer value of up to 18 digits, optionally preceded by a sign.

date is a date, in the format defined by your installation.

'*band-type*' is a text string of up to 78 characters, including delimiters.

Note:

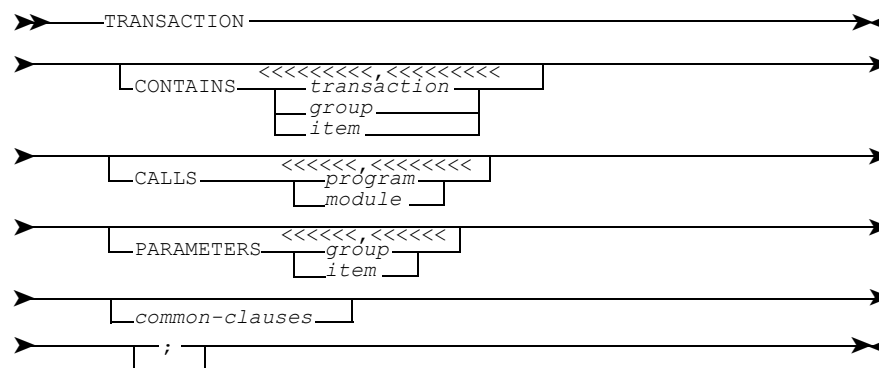
These delimiters are needed when defining a SYSTEM member via the command interface.

integer is as defined above.

common-clauses are any of the clauses common to all member types.

TRANSACTION

A specific set of input data that triggers the execution of a program.



where:

transaction is the name of a TRANSACTION member.

group is the name of a GROUP member.

item is the name of an ITEM member.

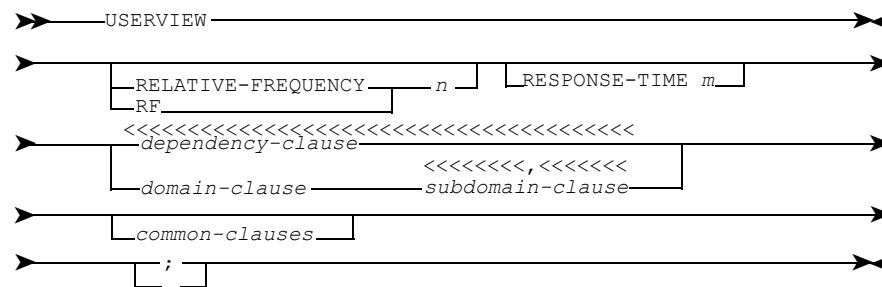
program is the name of a PROGRAM member.

module is the name of a MODULE member.

common-clauses are any of the clauses common to all member types.

USERVIEW

To define the access paths between data elements required by a user in a particular application-oriented view of data.

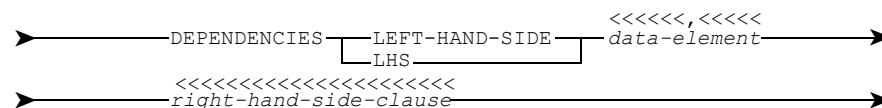


where:

n is an unsigned integer indicating the relative frequency of access to the USERVIEW being defined.

m is an unsigned integer indicating the response time required for data accessed by the process represented by the USERVIEW.

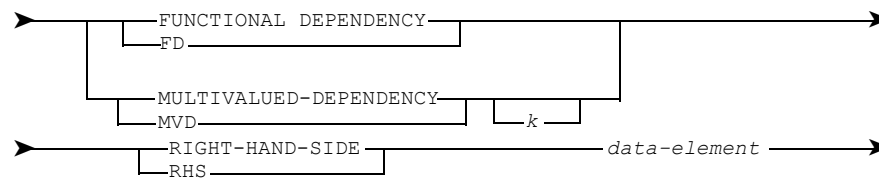
dependency-clause is:



where:

data-element is the name of a data element that is held in the modeling repository as an ITEM or GROUP member or is to be entered in the modeling repository as a dummy ITEM.

right-hand-side-clause is:

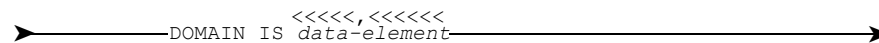


where:

k is an unsigned integer indicating the multiplicity for a multivalued dependency; that is, the average number of values (or sets of values) of the right-hand side of the dependency determined by a given value of the left-hand side. If not specified for a dependency, the default value of k is taken to be 1. (The multiplicity of a functional dependency is automatically taken to be 1.)

data-element is as defined above.

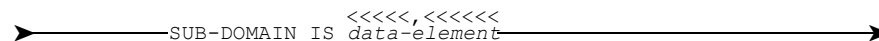
domain-clause is:



where:

data-element is as defined above.

subdomain-clause is:



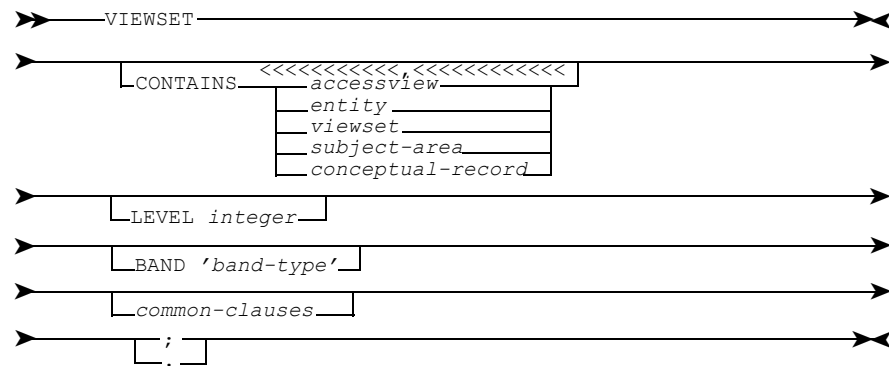
where:

data-element is as defined above.

common-clauses are any of the clauses common to all member types.

VIEWSET

To represent a collection of dataviews grouped into logically related application areas or subject areas.



where:

`accessview` is the name of an ACCESSVIEW member.

`entity` is the name of an ENTITY member.

`viewset` is the name of a VIEWSET member.

`subject-area` is the name of a SUBJECT-AREA member.

`conceptual-record` is the name of a conceptual-record member.

`integer` is an integer value of up to 18 digits, optionally preceded by a sign.

`'band-type'` is a text string of up to 78 characters, including delimiters.

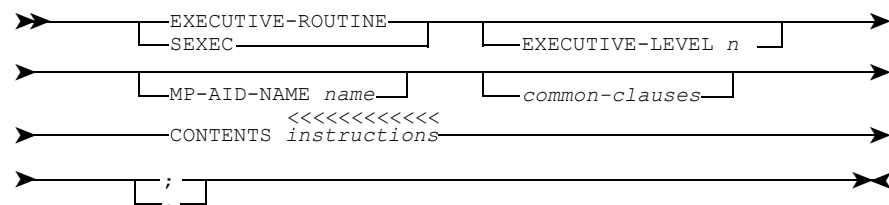
`common-clauses` are any of the clauses common to all member types.

3

Procedures Language

EXECUTIVE-ROUTINE and SEXEC MEMBER-TYPES Syntax

To define a Procedure.



where:

n is an integer from 0 to 255

name is a name not exceeding 10 characters in length

common-clauses are any of the clauses available to all member types

instructions comprise the Procedures Language instructions that make up the routine.

User Assigned Variables

Variable	Value
Global variables	&G0 to &G99
Local variables	&L0 to &L99
Installation variables	&I0 to &I99
Parameter variables	&P0 to &P99
Command variables	
Profile variables	

System Variables

&BUFN

Buffer name, if any, of the currently addressed buffer.

&CCOD

Return code from the previous command or executive routine.

&CCOL

Offset from the start of a data line to the character under which the cursor is placed, the first character being number zero. If the cursor is in the Line Command Area, the Command Area, or the Heading Area, &CCOL is set to -1.

&COLO

Offset of the first visible column of text at the left margin in the current buffer. If the buffer has not been offset due to a RIGHT command, the offset is given as zero.

&CROW

Number of data lines from the ***TOP OF DATA*** line to the data line where the cursor is placed. The value of &CROW is set to one for the first data line. If the cursor is in the Line Command Area, the Command Area, or the Heading Area, &CROW is set to -1.

&CURL

Contents of the current line in the currently addressed buffer.

&CURS

Value of an element of text or a delimited character string at the cursor position.

&DATE

The current date, in the format defined in the DCUST installation macro.

&DICT

Name of the currently open repository, if any.

&ECOD

The highest return code from executive routines or commands executed in the current executive routine.

&ENAM

Name of the current executive routine.

&ENVO

Current Operating System where 0 = OS, D = DOS, V = VM/CMS, and S = BS2000.

&ENVM

Environment in use where A = access call, F = full-screen interactive, L = line-mode interactive, and S = standard (batch) environment.

&ENVT

TP Monitor in use where A = access call interactive (when no other setting is appropriate), B = batch, C = CICS, F = ICCF, I = TSO/ISPF (only when ISPF is present), M = IMS/DC, R = ROSCOE, S = Siemens Timesharing Interface (TIAM), T = TSO, and V = VM/CMS.

&LINO

The number of data lines from the ***TOP OF DATA*** line to the current line in the currently addressed buffer, the first data line being number one.

&LOGO

The current Logon identifier.

&MODE

Buffer mode of currently addressed buffer where C = Command Mode, I = InfoBank Mode, L = Lookaside Mode, E = Edit Mode, and U = Update Mode.

&MSLN

Complete message line of the last Manager Products message.

&MSLV

Severity level of the last Manager Products message where I = Informatory message, W = Warning message, E = Error message, S = Serious error message, and C = Critical error message.

&MSNO

Number of the last Manager Products message.

&MSTX

Text of the last Manager Products message.

&PNUM

Number of input parameters supplied to the current executive routine.

&PVAL

Contents of all input parameters supplied to the current executive routine, exactly as entered by the user.

&SCOD

Highest return code so far of the current Manager Products session.

&STAT

Name of current status, if any.

&TIME

Current time, in the format defined in the DCUST installation macro.

&TRMC

Number of columns on the terminal screen.

&TRMR

Number of rows on the terminal screen.

&USER

Name of the current repository user, if any.

Return Codes

Return codes	0	-	I	Informatory message
	4	-	W	Warning message
	8	-	E	Error message
			S	Serious message
			C	Critical message

&CCOD

Return code from the previous command or executive routine.

&ECOD

The highest return code from executive routines or commands executed in the current executive routine.

&SCOD

Highest return code so far of the current Manager Products session.

EXPRESSIONS

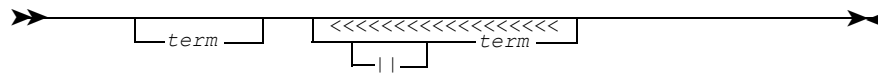
The maximum length of an expression is 255 characters.

These are operators recognized by Procedures Language:

	Concatenation (joining of character strings)
+	Addition
-	Subtraction
*	Multiplication
/	Integer division giving quotient
!	Integer division giving remainder
**	Raise to a power

Character Expressions

This is the definition of a character expression:



where:

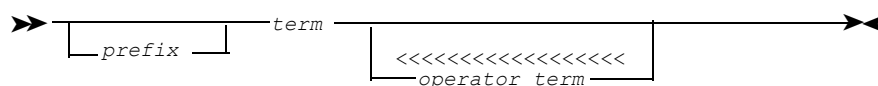
term is any of these:

- A variable name
- A function cell
- An unlimited character string with no embedded blanks
- Any delimited character string

|| is the concatenation operator.

Numeric Expressions

This is the definition of a numeric expression:



where:

term is any of these:

- A variable name containing an integer
- An integer
- A function

prefix is any of these:

- + (plus)
- - (minus)

operator is any of these:

- + (addition)
- - (subtraction)
- * (multiplication)
- / (integer division giving quotient)
- ! (integer division giving remainder)
- ** (raise to a power)

All operators have equal precedence. Processing is strictly from left to right.

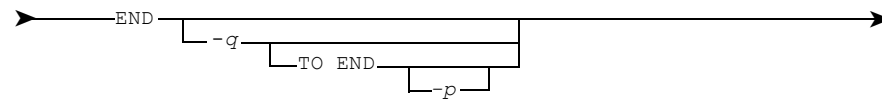
Expressions must be in the range -2147483648 to 2147483647, inclusive, except for the second operand of ** which must be in the range 0 to 15, inclusive.

Null strings are not considered to be numeric terms.

p and q are integers. The alphanumeric value of p must be less than the alphanumeric value of q .

string is a character string of between 1 and 32 printable characters.

name-end is:



where:

p and q are as defined above.

EQ (or =) means equal to.

NE means not equal to.

GT (or $>$) means greater than.

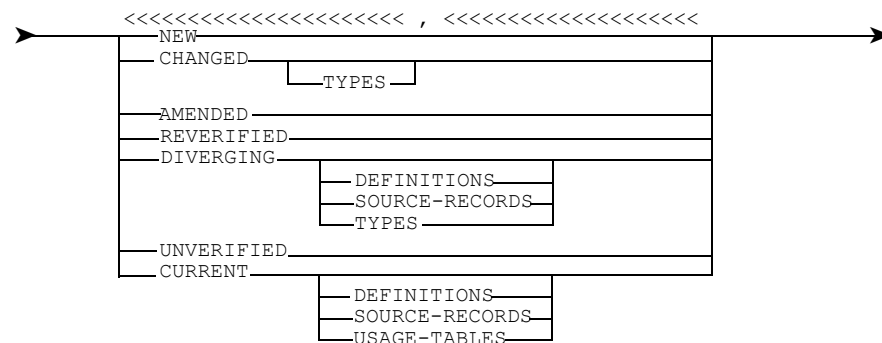
GE means greater than or equal to.

LT (or \leq) means less than or equal to.

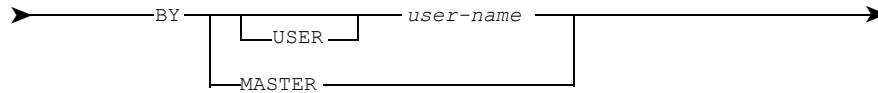
LE means less than or equal to.

integer is an unsigned integer in the range of 1 to 79, specifying a number of characters.

Secondary Selection Clauses: Status-related-selection



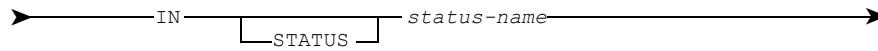




where:

user-name is the name of a user registered in the repository.

status is:

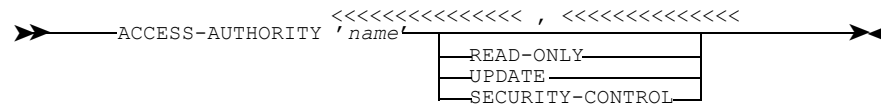


where *status-name* is the name of a status visible from the current status.

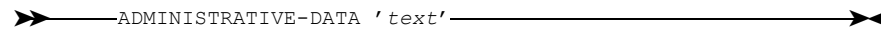
Appendix B

Common Clauses

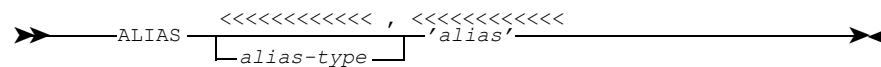
Common clauses are any of:



where *name* is a name not more than 32 characters in length.



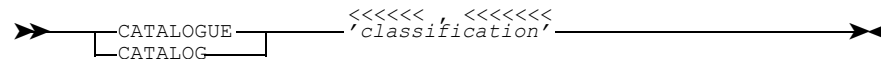
where *text* is up to 32767 delimited character strings, each string having a maximum of 246 characters.



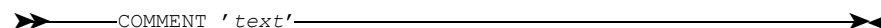
where:

alias is the alternative name for the member and must not be more than 79 printable characters.

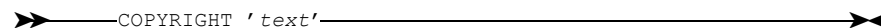
alias-type is one of the specific alias-type keywords available in your dictionary. Use the `SHOW ALIAS-TYPES` command to find out the alias-type keywords available in your repository.



where *classification* is a string of not more than 79 characters.



where *text* is as defined above.



where *text* is as defined above.

➤➤——DESCRIPTION '*text*'———➤➤

where *text* is as defined above.

➤➤——EFFECTIVE-DATE *date*———➤➤

where *date* must be in the format which applies in your installation.

➤➤——FREQUENCY *frequency*———➤➤

ACCESS	frequency
RUN	
UPDATE	
BACKUP	

where *frequency* is number.

➤———➤

number	
' <i>string</i> '	
MONTHLY	
WEEKLY	
DAILY	
HOURLY	number
	' <i>string</i> '

where:

number is an integer.

string is a string of not more than 256 characters.

➤➤——NOTE '*text*'———➤➤

where *text* is as defined above.

➤➤——OBSOLETE-DATE *date*———➤➤

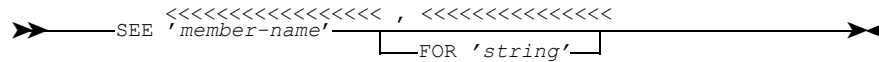
where *date* is as defined above.

➤➤——QUERY '*text*'———➤➤

where *text* is as defined above.

➤➤——SECURITY-CLASSIFICATION '*string*' , '*string*' FROM *date*———➤➤

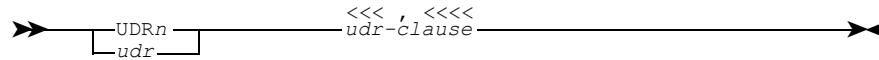
where *string* and *date* are as defined above.



where:

member-name is the name of a dictionary member.

string is as defined above.



where:

n is an integer in the range 1 to 9.

udr is the name of a user-defined EA relationship type.

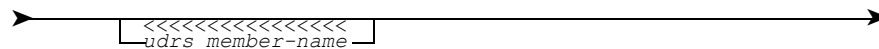
udr-clause is:



where:

member-name is the name of a dictionary member.

string is as defined above.



Appendix C

Features Table

This table lists:

- All the commands, member types, and procedures language instructions documented in this manual.
- The functional and selectable unit codes of the ASG-MethodManager and non-ASG-MethodManager products they are supplied with.

Keywords within a command are also listed if they are supplied with an additional selectable or functional unit.

To find out:

- The commands available in your environment use the QUERY PRIMARY and QUERY LINE-COMMAND commands
- The Manager Products available in your environment use the ENVIRONMENT command or ASG-MethodManager panel TZ80000
- The member types available in your environment use the SHOW UDS command.

Name	FU Code(s)	SU Code(s)
+	MMR-FW20	No equivalent
*	MMR-FW20	CMR-FE01
/	MMR-FW20	CMR-FE01
?	MMR-FW20	No equivalent
??DUP	MMR-FW20	No equivalent
??EXTRACT	MMR-FW20	No equivalent
??INCLUDE	MMR-FW20	No equivalent
??LINE	MMR-FW20	No equivalent
??LMAR	MMR-FW20	No equivalent
??PAGE	MMR-FN20	No equivalent

Name	FU Code(s)	SU Code(s)
??PBOT	MMR-FN20	No equivalent
??PTOP	MMR-FN20	No equivalent
??RMAR	MMR-FN20	No equivalent
??SKIP	MMR-FN20	No equivalent
??TITLE	MMR-FN20	No equivalent
=	MMR-FW20	No equivalent
"	MMR-FW20	CMR-FE01
A	MMR-FW20	CMR-FE01
ABBREV	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
ABEND	MMR-FW30	CMR-CM01
ACCEPT	MMR-FW20	No equivalent
ACT	MMR-FW10	No equivalent
ACTIVATE	MMR-FW10	No equivalent
ADD	MMR-FW30	CMR-CM01
ADIS	MMR-FW20	No equivalent
ADISPLAY	MMR-FW20	No equivalent
ADW EXPORT	MMR-PF72 and MMR- FN27	DYR-TE15
ADW IMPORT	MMR-PF72 and MMR- FN28	DYR-TI15
AGET	MMR-FW20	No equivalent
AGETC	MMR-FW20	No equivalent
ALSO KEEP	MMR-PF61	DMR-DD1
ALTER	MMR-FW30	CMR-CM01
AMEND	MMR-FW30	CMR-CM01
ARG	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
ARRAYGEN	MMR-FW30	DYR-TE00
ARRAYHI	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05

Name	FU Code(s)	SU Code(s)
ARRAYLO	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
ASM	MMR-PF33 and MMR-FN27	DMR-SL3
AUPD	MMR-FW20	No equivalent
AUPDATE	MMR-FW20	No equivalent
AUTHORITY	MMR-FW30	CMR-CM01
B	MMR-FW20	No equivalent
BACKWARD	MMR-FW30	CMR-CM01
BOTTOM	MMR-FW30	CMR-CM01
BROWSE	MMR-FW20	CMR-FE01
BS2000	MMR-PF44	CMR-TP11
BS2000 ACTIVATE CURSOR-CONTROL	MMR-PF44	CMR-TP11
BS2000 ACTIVATE P-KEY-PROTECTION	MMR-PF44	CMR-TP11
BS2000 ACTIVATE STXIT-TRACE	MMR-PF44	CMR-TP11
BS2000 BREAKPOINT	MMR-PF44	CMR-TP11
BS2000 DEACTIVATE CURSOR-CONTROL	MMR-PF44	CMR-TP11
BS2000 DEACTIVATE P-KEY-PROTECTION	MMR-PF44	CMR-TP11
BS2000 DEACTIVATE STXIT-TRACE	MMR-PF44	CMR-TP11
BS2000 EXECUTE &EDT	MMR-PF44	CMR-TP11
BS2000 EXECUTE &LMS	MMR-PF44	CMR-TP11
BS2000 HELP	MMR-PF44	CMR-TP11
BS2000 QUERY CORE	MMR-PF44	CMR-TP11
BS2000 QUERY ENQUEUE-TABLES	MMR-PF44	CMR-TP11
BS2000 QUERY ERRORS	MMR-PF44	CMR-TP11
BS2000 QUERY FILES	MMR-PF44	CMR-TP11

Name	FU Code(s)	SU Code(s)
BS2000 QUERY MODULES	MMR-PF44	CMR-TP11
BS2000 RESET P-KEYS	MMR-PF44	CMR-TP11
BS2000 RESET EDT-AREAS	MMR-PF44	CMR-TP11
BS2000 SET P-KEYS	MMR-PF44	CMR-TP11
BUILD	MMR-FW30	DYR-TE00
BULK	MMR-PF61	DMR-DD1
ALL-STATUSES	MMR-FN30	CMR-DD2 or CMR-AD21
AMENDED	MMR-FN30	MMR-DD2 or CMR-AD21
CHANGED	MMR-FN30	CMR-DD2 or CMR-AD21
CHECK	MMR-PF61	No equivalent
CHECK-NEEDED	MMR-PF61	No equivalent
CHECK-OK	MMR-PF61	No equivalent
CURRENT	MMR-FN30	CMR-DD2 or CMR-AD21
DIVERGING	MMR-FN30	CMR-DD2 or CMR-AD21
GENERIC	MMR-FW20	CMR-UD1
LOCKED	MMR-FW30	CMR-WS01
NEW	MMR-FN30	CMR-DD2 or CMR-AD21
UNVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
REVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
C	MMR-FW20	CMR-FE01
CALCULATE	MMR-FN24	DSR-PH10
CALL	MMR-FW30 or MMR-FW20	DYR-TE00
CANCEL	MMR-FW20	No equivalent
CC	MMR-FW20	CMR-FE01
CENTER	MMR-FW30 or MMR-FW20	DYR-TE00
CENTRE	MMR-FW30 or MMR-FW20	DYR-TE00

Name	FU Code(s)	SU Code(s)
CEEXEC	MMR-FW30 or MMR-FW20	DYR-TE00
CHANGE	MMR-FW20	CMR-FE01
CHECK	MMR-PF61	No equivalent
CLEAR	MMR-FW30	CMR-CM01
CLOSE	MMR-FW30	CMR-CM01
CLOSEF	MMR-FW30	DYR-TE00
CMS	MMR-PF48	CMR-TP8
COB	MMR-PF31 and MMR-FN27	No equivalent
COBOL	MMR-PF31 and MMR-FN27	No equivalent
COMMAND	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
CONVERT		
COBOL	MMR-PF31 and MMR-FN28	DMR-AS1
PL/I	MMR-PF32 and MMR-FN28	DMR-AS2
COPIES	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
COPY	MMR-FW30	CMR-CM01
FROM	MMR-FN30	CMR-DD2 or CMR-AD21
COPYRIGHT	MMR-FW30	CMR-CM01
CORPORATE- EXECUTIVE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
D	MMR-FW20	CMR-FE01
DACCESS	MMR-FW30	DYR-TE00
DACT	MMR-FW10	No equivalent
DB2 ALTER	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 BIND	MMR-PF21 and MMR-FN27	DYR-TE12

Name	FU Code(s)	SU Code(s)
DB2 COMMENT	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 CREATE	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 DEBUG	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 DECLARE	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 DROP	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 GRANT	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 LABEL	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 LIST CYCLES	MMR-FN24	DSR-PH02
DB2 LIST TABLES	MMR-FN24	DSR-PH02
DB2 PLOT CLUSTER	MMR-FN24	DSR-PH02
DB2 PLOT REFERENTIAL- STRUCTURES	MMR-FN24	DSR-PH02
DB2 POPULATE	MMR-FN24	DSR-PH02
USING FORMAT	MMR-FN24	DSR-UD30
DB2 PREVIEW	MMR-FN24	DSR-PH02
USING FORMAT	MMR-FN24	DSR-UD30
DB2 PRODUCE	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 REBIND	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 RECALCULATE	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 REPORT	MMR-FN24	DSR-PH02
USING FORMAT	MMR-FN24	DSR-UD30
DB2 REVOKE	MMR-PF21 and MMR-FN27	DYR-TE12

Name	FU Code(s)	SU Code(s)
DB2 SIZE	MMR-PF21 and MMR-FN27	DYR-TE12
DB2 SYNONYM	MMR-PF21 and MMR-FN27	DYR-TE12
DB2-ALIAS	MMR-PF21	DMR-DD12
DB2-DATABASE	MMR-PF21	DMR-DD12
DB2-INDEX	MMR-PF21	DMR-DD12
DB2-LOCATION	MMR-PF21	DMR-DD12
DB2-PRIVILEGE	MMR-PF21	DMR-DD12
DB2-STOGROUP	MMR-PF21	DMR-DD12
DB2-TABLE	MMR-PF21	DMR-DD12
DB2-TBSPACE	MMR-PF21	DMR-DD12
DB2-USER	MMR-PF21	DMR-DD12
DB2-VIEW	MMR-PF21	DMR-DD12
DCUPD	MMR-FN20	No equivalent
DD	MMR-FW20	CMR-FE01
DEACTIVATE	MMR-FW10	No equivalent
DELETE	MMR-FW20	CMR-FE01
DESIGN	MMR-FN24	DSR-DS01
USING FORMAT	MMR-FN24	DSR-UD30
DEXPAND	MMR-FW30	DYR-TE00
DIAGRAM	MMR-FW30	CMR-WS01
DICTIONARY	MMR-FW30	CMR-CM01
DIS	MMR-FW20	No equivalent
DISPLAY	MMR-FW20	No equivalent
DL/I LIST	MMR-FN24	DSR-PH05
DL/I MAP	MMR-FN24	DSR-PH05
DL/I REPORT	MMR-FN24	DSR-PH05
DL/I-DATABASE GSAM	MMR-PF24	DMR-DD5
DL/I-DATABASE HDAM	MMR-PF24	DMR-DD5

Name	FU Code(s)	SU Code(s)
DL/I-DATABASE HIDAM	MMR-PF24	DMR-DD5
DL/I-DATABASE HISAM	MMR-PF24	DMR-DD5
DL/I-DATABASE HSAM	MMR-PF24	DMR-DD5
DL/I-DATABASE INDEX	MMR-PF24	DMR-DD5
DL/I-DATABASE LOGICAL	MMR-PF24	DMR-DD5
DL/I-DATABASE SECONDARY-INDEX	MMR-PF24	DMR-DD5
DMS	MMR-FW30	CMR-CM01
DO	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
DOC	MMR-FN10	No equivalent
DOES	MMR-PF61	DMR-DD1
DOWN	MMR-FW20	CMR-FE01
DRELEASE	MMR-FW30	DYR-TE00
DRETRIEVE	MMR-FW30	DYR-TE00
DROP command	MMR-PF61	DMR-DD1
DROP directive	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
EDDATE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
EDIT	MMR-FW20	CMR-FE01
EDTIME	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
ENCODE	MMR-FW30	CMR-CM01
END command	MMR-FW30	CMR-CM01
END directive	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
ENVIRONMENT	MMR-FW30	CMR-CM01
ERASE KEPT-DATA	MMR-FW30	CMR-CM01
EXECUTE	MMR-FW20	CMR-FE01
EXIT	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05

Name	FU Code(s)	SU Code(s)
EXTRACT command	MMR-FW30	DYR-TE00
EXTRACT function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
EXTRACT DB2	MMR-PF21 and MMR-FN28	DYR-TI12
EXTRACT SQL	MMR-PF22 and MMR-FN28	DYR-TI32
F	MMR-FW20	CMR-FE01
FDO	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
FETCH KEPT-DATA	MMR-FW30	CMR-CM01
FETCH WBDA MMR	MMR-FN24	DSR-DS01
FILE	MMR-FW20	CMR-FE01
FIND	MMR-FW20	CMR-FE01
FLUSH	MMR-FN24	DSR-DS01
FORMAT		
AS	MMR-FW30	CMR-CM01
VIA	No equivalent	DYR-UD15
FORWARD	MMR-FW30	CMR-CM01
FREE	MMR-FW30	CMR-CM01
GETD	MMR-FW20	CMR-FE01
GETE	MMR-FW20	CMR-FE01
GETS	MMR-FW20	CMR-FE01
GETT	MMR-FW20	CMR-FE01
GETU	MMR-FW20	CMR-FE01
GET-DICTIONARY	MMR-FW20	CMR-FE01
GET-EXECUTIVE	MMR-FW20	CMR-FE01
GET-SCRATCHPAD	MMR-FW20	CMR-FE01
GET-TRANSIENT	MMR-FW20	CMR-FE01
GET-USER	MMR-FW20	CMR-FE01
GLOBAL	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05

Name	FU Code(s)	SU Code(s)
GLOSSARY	MMR-PF61	DMR-DD1
ALL-STATUSES	MMR-FN30	CMR-DD2 or CMR-AD21
AMENDED	MMR-FN30	CMR-DD2 or CMR-AD21
CHANGED	MMR-FN30	CMR-DD2 or CMR-AD21
CHECK-NEEDED	MMR-PF61	No equivalent
CHECK-OK	MMR-PF61	No equivalent
CURRENT	MMR-FN30	CMR-DD2 or CMR-AD21
DIVERGING	MMR-FN30	CMR-DD2 or CMR-AD21
GENERIC	MMR-FW20	CMR-UD1
LOCKED	MMR-FW30	CMR-WS01
NEW	MMR-FN30	CMR-DD2 or CMR-AD21
UNVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
REVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
GOTO	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
HARDCOPY	MMR-FW20	No equivalent
HELP	MMR-FW30	CMR-CM01
HOLD	MMR-FW20	CMR-FE01
I	MMR-FW20	CMR-FE01
IDMS-AREA	No equivalent	DMR-DD8
IDMS-DATABASE	No equivalent	DMR-DD8
IDMS-LOGICAL- RECORD	No equivalent	DMR-DD8
IDMS-PATH-GROUP	No equivalent	DMR-DD8
IDMS-RECORD	No equivalent	DMR-DD8
IDMS-SET	No equivalent	DMR-DD8
IDMS-SUBSCHEMA	No equivalent	DMR-DD8
IDMS-VIEW	No equivalent	DMR-DD8
IEW EXPORT	MMR-PF71 and MMR-FN27	DYR-TE14

Name	FU Code(s)	SU Code(s)
IEW IMPORT	MMR-PF71 and MMR-FN28	DYR-TE14
IF	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
IMS LIST	MMR-FN24	DSR-PH05
IMS MAP	MMR-FN24	DSR-PH05
IMS REPORT	MMR-FN24	DSR-PH05
IMS-DATABASE GSAM	MMR-PF24	DMR-DD5
IMS-DATABASE HDAM	MMR-PF24	DMR-DD5
IMS-DATABASE HIDAM	MMR-PF24	DMR-DD5
IMS-DATABASE HISAM	MMR-PF24	DMR-DD5
IMS-DATABASE HSAM	MMR-PF24	DMR-DD5
IMS-DATABASE INDEX	MMR-PF24	DMR-DD5
IMS-DATABASE LOGICAL	MMR-PF24	DMR-DD5
IMS-DATABASE SECONDARY-INDEX	MMR-PF24	DMR-DD5
INFOBANK	MMR-FW30	CMR-CM01
INFO-BANK	MMR-FW30	CMR-CM01
INSERT command	MMR-FW30	CMR-CM01
INSERT function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
INTERPRET	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
ISPF	MMR-PF47	CMR-TP7
ISQL	MMR-FW30	DYR-TE00
ITERATE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
JUSTIFY	MMR-FW20	CMR-FE01
K	MMR-FW20	CMR-FE01
KK	MMR-FW20	CMR-FE01
KEEP	MR-PF61	DMR-DD1

Name	FU Code(s)	SU Code(s)
KPUT	MMR-FW20	CMR-FE01
LADD	MMR-FW20	CMR-FE01
LASTPOS	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
LCS	MMR-FW10	No equivalent
LEAVE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
LEFT command	MMR-FW30	CMR-CM01
LEFT function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
LENGTH	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
LIST	MMR-FW30	CMR-CM01
ALL-STATUSES	MMR-FN30	CMR-DD2 or CMR-AD21
AMENDED	MMR-FN30	CMR-DD2 or CMR-AD21
CHANGED	MMR-FN30	CMR-DD2 or CMR-AD21
CHECK-NEEDED	MMR-PF61	No equivalent
CHECK-OK	MMR-PF61	No equivalent
CURRENT	MMR-FN30	CMR-DD2 or CMR-AD21
DIVERGING	MMR-FN30	CMR-DD2 or CMR-AD21
GENERIC	MMR-FW20	CMR-UD1
LOCK-INFORMATION	MMR-FW30	CMR-WS01
LOCKED	MMR-FW30	CMR-WS01
LOCKS	MMR-FW30	CMR-WS01
NEW	MMR-FN30	CMR-DD2 or CMR-AD21
REVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
UNVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
LIST data modeling and design	MMR-FN24	DSR-DS01
LITERAL	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05

Name	FU Code(s)	SU Code(s)
LOCAL	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
LOCATE	MMR-FW20	CMR-FE01
LOCK	MMR-FW30	CMR-WS01
LOGOFF	MMR-FW30	CMR-CM01
LOGON	MMR-FW30	CMR-CM01
LOOKASIDE	MMR-FW20	CMR-FE01
LOWER	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
M	MMR-FW20	CMR-FE01
MAIL	MMR-FW30	CMRUI1
MATRIX	MMR-PF61	No equivalent
MEMTYPE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MER	MMR-FN24	No equivalent
MERE	MMR-FN24	No equivalent
MERGE		
COBOL	MMR-PF31 and MMR-FN28	DMR-AS1
PL/I	MMR-PF32 and MMR-FN28	DMR-AS2
MERGE data modeling and design	MMR-FN24	DSR-DS01
MERU	MMR-FN24	DSR-DS01
MERUE	MMR-FN24	DSR-DS01
MESSAGE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MM	MMR-FW20	CMR-FE01
MODE		
DMS	MMR-FW30	CMR-CM01
WBDA	MMR-FN24	DSR-DS01
MODIFY	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
MOVE	MMR-FW20	CMR-FE01
MP-AID CONCATENATION LIST	MMR-FW30	CMR-CM01
MP-AID DELETE	MMR-FW30	CMR-CM01
MP-AID EXEC-LIST	MMR-FW30	CMR-CM01
MP-AID LIST	MMR-FW30	CMR-CM01
MP-AID PASSWORD	MMR-FW30	CMR-CM01
MP-AID PRINT	MMR-FW30	CMR-CM01
MPR	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPRAID	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPRCMPW	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPRDDPW	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPRSU	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPRUCLS	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPRUDSN	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MPXX	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
MTHelp	MMR-FW20	No equivalent
MVW	MMR-FW30	CMR-WS01
NAME	MMR-FN24	DSR-DS01
NDATE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
NEXT	MMR-FW20	CMR-FE01
NOP	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
NOPRINT	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
NTIME	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
OVERLAY	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
P	MMR-FW20	CMR-FE01
PANEL	MMR-FW30	CMR-CM01
PARSABLE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
PARSE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
PARSEOPTION	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
PERFORM	MMR-PF61	DMR-DD1
AMENDED	MMR-FN30	CMR-DD2 or CMR-AD21
CHANGED	MMR-FN30	CMR-DD2 or CMR-AD21
CHECK-NEEDED	MMR-PF61	No equivalent
CHECK-OK	MMR-PF61	No equivalent
CURRENT	MMR-FN30	CMR-DD2 or CMR-AD21
DIVERGING	MMR-FN30	CMR-DD2 or CMR-AD21
GENERIC	MMR-FW20	CMR-UD1
LOCKED	MMR-FW30	CMR-WS01
NEW	MMR-FN30	CMR-DD2 or CMR-AD21
UNVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
REVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
PL/I	MMR-FN27 and MMR-PF32	
PLOT	MMR-FN24	DSR-UD31
POPULATE		
ENTITIES	MMR-FN24	DSR-DS01
FROM	MMR-FW30	DYR-TE00
USERVIEWS	MMR-FN24	DSR-DS01
USING FORMAT	MMR-FN24	DSR-UD30

Name	FU Code(s)	SU Code(s)
POS	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
POST	MMR-FW30	CMR-UI1
PREVIEW		
ENTITIES	MMR-FN24	DSR-DS01
IMPORT	MMR-FW30	DYR-TE00
USERVIEWS	MMR-FN24	DSR-DS01
USING FORMAT	MMR-FN24	DSR-UD30
PRINT	MMR-FW30	CMR-CM01
ALL-STATUSES	MMR-FN30	CMR-DD2 or CMR-AD21
PRODUCE		
ASSEMBLER	MMR-PF33 and MMR-FN27	DMR-SL3
COBOL	MMR-PF31 and MMR-FN27	DMR-SL1
PL/I	MMR-PF32 and MMR-FN27	DMR-SL2
PRODUCE DB2	MMR-FN24	DSR-PH02
PRODUCE DL/I DBDGEN	MMR-PF25 and MMR-FN27	DMR-SL5
PRODUCE DL/I PSBGEN	MMR-PF25 and MMR-FN27	DMR-SL5
PRODUCE IMS DBDGEN	MMR-PF24 and MMR-FN27	DMR-SL5
PRODUCE IMS PSBGEN	MMR-PF24 and MMR-FN27	DMR-SL5
PRODUCE SQL	MMR-FN24	DSR-PH01
PROJLIST	MMR-FN10	No equivalent
PROJVIEW	MMR-FN10	No equivalent
PROTECT	MMR-FW30	CMR-DD3
PTIME	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
PULL	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
PUR	MMR-FW20	No equivalent
PURGE	MMR-FW20	No equivalent
PUSH	MMR-FW30	CMR-CM01
PUT	MMR-FW20	CMR-FE01
Q	MMR-FW20	CMR-FE01
QQ	MMR-FW20	CMR-FE01
QUERY ABEND-MESSAGE	MMR-FW30	CMR-CM01
QUERY ACTIVE-BUFFERS	MMR-FW30	CMR-CM01
QUERY ALERT	MMR-FW30	CMR-CM01
QUERY ALL	MMR-FW30	CMR-CM01
QUERY AMBIGUITY- ASSUMPTION	MMR-FW30	CMR-CM01
QUERY AUTOSKIP	MMR-FW30	CMR-CM01
QUERY BLANK-LINE-DISPLAY	MMR-FW30	CMR-CM01
QUERY BROWSE-EDIT	MMR-FW30	CMR-CM01
QUERY BROWSE-RETENTION	MMR-FW30	CMR-CM01
QUERY BUFFER-LIMIT	MMR-FW30	CMR-CM01
QUERY CEI MMR	MMR-FW30	CMR-CM01
QUERY CHARACTER- TRANSLATION	MMR-FW30	CMR-CM01
QUERY CHECK-CHARACTER	MMR-FW30	CMR-CM01
QUERY CMI	MMR-FW30	CMR-CM01
QUERY COMMAND-LINE	MMR-FW30	CMR-CM01
QUERY COMMAND-MEMBER- INDEX	MMR-FW30	CMR-CM01
QUERY COMMAND- RETENTION	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
QUERY CORPORATE-EXECUTIVE-INDEX	MMR-FW30	CMR-CM01
QUERY COUNTER	MMR-FW30	CMR-CM01
QUERY CURSOR-HOLD	MMR-FW30	CMR-CM01
QUERY DICTIONARY	MMR-FW30	CMR-CM01
QUERY DICTIONARY-UPDATES	MMR-FW30	CMR-CM01
QUERY DIRECTLY-ASSUMPTION	MMR-FW30	CMR-CM01
QUERY DISABLED-MPAID-CHECK	MMR-FW30	CMR-CM01
QUERY DBCS-MODE	MMR-FW30	CMR-CM01
QUERY DBCS-STRING-DELIMITER	MMR-FW30	CMR-CM01
QUERY DSR	MMR-FW30	CMR-CM01
QUERY ECHO	MMR-FW30	CMR-CM01
QUERY ENQUEUE-TRACE	MMR-FW30	CMR-CM01
QUERY ENTER-KEY	MMR-FW30	CMR-CM01
QUERY ESCAPE-CHARACTER	MMR-FW30	CMR-CM01
QUERY EXCP-LIMIT	MMR-FW30	CMR-CM01
QUERY EXCP-MONITOR	MMR-FW30	CMR-CM01
QUERY EXEC-WRITE	MMR-FW30	CMR-CM01
QUERY EXECUTIVE-RETENTION	MMR-FW30	CMR-CM01
QUERY FORMAT	MMR-FW30	CMR-CM01
QUERY FORMAT-TITLE	MMR-FW30	CMR-CM01
QUERY FREE-POOL	MMR-FW30	CMR-CM01
QUERY HELP-ENTRY	MMR-FW30	CMR-CM01
QUERY INDEX-PANEL	MMR-FW30	CMR-CM01
QUERY INFOBANK-ENTRY	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
QUERY INFOBANK-PANEL-INDEX	MMR-FW30	CMR-CM01
QUERY INFOBANK-SELECT	MMR-FW30	CMR-CM01
QUERY INTERROGATE-ENQUEUE	MMR-FW30	CMR-CM01
QUERY IO-FLUSH-LIMIT	MMR-FW30	CMR-CM01
QUERY IPI	MMR-FW30	CMR-CM01
QUERY KANJI-MODE	MMR-FW30	CMR-CM01
QUERY KANJI-STRING-DELIMITER	MMR-FW30	CMR-CM01
QUERY KEPT-DATA	MMR-FW30	CMR-CM01
QUERY LA-RETENTION	MMR-FW30	CMR-CM01
QUERY LINE-AREA	MMR-FW30	CMR-CM01
QUERY LINE-COMMANDS	MMR-FW30	CMR-CM01
QUERY LINE-END-CHARACTER	MMR-FW30	CMR-CM01
QUERY LINEAR-RETENTION	MMR-FW30	CMR-CM01
QUERY LIST-SHIFT	MMR-FW30	CMR-CM01
QUERY LOADLIST	MMR-FW30	CMR-CM01
QUERY LOGON-ID	MMR-FW30	CMR-CM01
QUERY LOOKASIDE-RETENTION	MMR-FW30	CMR-CM01
QUERY LUW	MMR-FW30	CMR-CM01
QUERY MESSAGE-LANGUAGE	MMR-FW30	CMR-CM01
QUERY MODE	MMR-FW30	CMR-CM01
QUERY MPAID-UPDATES	MMR-FW30	CMR-CM01
QUERY NUMBERS	MMR-FW30	CMR-CM01
QUERY ONLINE-USERS	MMR-FW30	CMR-CM01
QUERY OUTPUT-EDIT	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
QUERY OUTPUT-LINE-LIMIT	MMR-FW30	CMR-CM01
QUERY PAGE-HEADINGS	MMR-FW30	CMR-CM01
QUERY PAGE-LENGTH	MMR-FW30	CMR-CM01
QUERY PERFORM-CHARACTER	MMR-FW30	CMR-CM01
QUERY PFKEYS	MMR-FW30	CMR-CM01
QUERY PREFIX-AREA	MMR-FW30	CMR-CM01
QUERY PREFIX-COMMANDS	MMR-FW30	CMR-CM01
QUERY PRIMARY-COMMANDS	MMR-FW30	CMR-CM01
QUERY PRINT-DELIMITERS	MMR-FW30	CMR-CM01
QUERY PRINT-SHIFT	MMR-FW30	CMR-CM01
QUERY RESERVED- INFOBANK-LINE	MMR-FW30	CMR-CM01
QUERY RETAINED- EXECUTIVES	MMR-FW30	CMR-CM01
QUERY REVISION-BARS	MMR-FW30	CMR-CM01
QUERY SCALE	MMR-FW30	CMR-CM01
QUERY SCREEN-REFRESH	MMR-FW30	CMR-CM01
QUERY SCROLL-LIMIT	MMR-FW30	CMR-CM01
QUERY SEGMENT	MMR-FW30	CMR-CM01
QUERY SOS-DUMP	MMR-FW30	CMR-CM01
QUERY SPACE-LINE-DISPLAY	MMR-FW30	CMR-CM01
QUERY STORAGE	MMR-FW30	CMR-CM01
QUERY STRING-DELIMITER	MMR-FW30	CMR-CM01
QUERY SUBSTITUTION- CHARACTER	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
QUERY TAB-CHARACTER	MMR-FW30	CMR-CM01
QUERY TABS	MMR-FW30	CMR-CM01
QUERY TOP-MENU-ENTRY	MMR-FW30	CMR-CM01
QUERY TRANLSATION	MMR-FW30	CMR-CM01
QUERY UMI	MMR-FW30	CMR-CM01
QUERY UPDATE-OUTPUT	MMR-FW30	CMR-CM01
QUERY UPPER-CASE	MMR-FW30	CMR-CM01
QUERY USER-BLOCKS	MMR-FW30	CMR-CM01
QUERYS USER- DIRECTORY-SEARCH	MMR-FW30	CMR-CM01
QUERY USER-MEMBER-INDEX	MMR-FW30	CMR-CM01
QUERY VERIFY	MMR-FW30	CMR-CM01
QUERY VIRTUAL	MMR-FW30	CMR-CM01
QUIT	MMR-FW20	CMR-FE01
R	MMR-FW20	No equivalent
RADD	MMR-FW30	DYR-TE00
READ	MMR-FW20	CMR-FE01
RECONCILE	MMR-FW30	DYR-TE00
REDUCE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
REF	MMR-FW20	No equivalent
REFA	MMR-FW20	No equivalent
REFERENCES	MMR-FW20	No equivalent
REINSTATE	MMR-FW30	CMR-CM01
RELEASE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
RELINQUISH	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
REMOVE	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
RENAME	MMR-PF61	
REPLACE	MMR-FW30	CMR-CM01
REPORT	MMR-FW30	CMR-CM01
ALL-STATUSES	MMR-FN30	CMR-DD2 or CMR-AD21
REPORT data modeling and design	MMR-FN24	DSR-DS01
USING FORMAT	MMR-FN24	DSR-UD30
REPSTR	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
RESERVE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
RESTART	MMR-FW30	CMR-CM01
RETAIN	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
RETRACE	MMR-FW30	CMR-CM01
RETURN command	MMR-FW30	CMR-CM01
RETURN directive	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
REVERSE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
REVERT	MMR-FN30	CMR-AD21 or CMR-DD2
RFILE	MMR-FW20	CMR-FE01
RIGHT command	MMR-FW30	CMR-CM01
RIGHT function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
RIGN	MMR-FW30	DYR-TE00
ROOT	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
RREN	MMR-FW30	DYR-TE00
RREP	MMR-FW30	DYR-TE00
RUN	MMR-FW20	CMR-FE01
RUPD	MMR-FW30	DYR-TE00
S	MMR-FW20	CMR-FE01

Name	FU Code(s)	SU Code(s)
SAY	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
SEARCH	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
SELECT	MMR-FW30	CMR-CM01
SENDF	MMR-FW30	DYR-TE00
SET directive	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
SET ALERT	MMR-FW20	CMR-FE01
SET AUTOSKIP	MMR-FW20	CMR-UD05
SET BLANK-LINE-DISPLAY	MMR-FW30	CMR-CM01
SET BROWSE-EDIT	MMR-FW20	CMR-UD05
SET BROWSE-RETENTION	MMR-FW20	CMR-UD05
SET CHECK-CHARACTER	MMR-FW30	CMR-CM01
SET COMMAND-LINE	MMR-FW30	CMR-CM01
SET COMMAND-RETENTION	MMR-FW30	CMR-CM01
SET COUNTER	MMR-FW30	CMR-CM01
SET CURSOR-HOLD	MMR-FW30	CMR-CM01
SET DIAGNOSTIC-DUMP	MMR-FW30	CMR-CM01
SET DIRECTLY- ASSUMPTION	MMR-FW30	CMR-CM01
SET DOCUWARE	MMR-FW30	CMR-CM01
SET ECHO	MMR-FW30	CMR-CM01
SET ENQUEUE-TRACE	MMR-FW30	CMR-CM01
SET ENTER-KEY	MMR-FW30	CMR-CM01
SET ESCAPE-CHARACTER	MMR-FW30	CMR-CM01
SET EXCP-MONITOR	MMR-FW20	CMR-FE01
SET EXEC-WRITE	MMR-FW20	CMR-UD05

Name	FU Code(s)	SU Code(s)
SET EXECUTIVE-RETENTION	MMR-FW30	CMR-CM01
SET FORMAT	MMR-FW30	CMR-CM01
GENERIC	MMR-FW20	CMR-UD1
TO	No equivalent	CMR-UD15
SET FORMAT-TITLE	No equivalent	DYR-UD15
SET INFOBANK-SELECT	MMR-FW30	CMR-CM01
SET LA-RETENTION	MMR-FW20	CMR-UD05
SET LINE-AREA	MMR-FW30	CMR-CM01
SET LINE-END-CHARACTER	MMR-FW30	CMR-CM01
SET LINEAR-RETENTION	MMR-FW20	CMR-UD05
SET LIST-SHIFT	MMR-FW30	CMR-CM01
SET LOOKASIDE-RETENTION	MMR-FW20	CMR-FE01
SET NUMBERS	MMR-FW20	CMR-FE01
SET OUPUT-EDIT	MMR-FW20	CMR-UD05
SET PAGE-HEADINGS	MMR-FW30	CMR-CM01
SET PAGE-LENGTH	MMR-FW30	CMR-CM01
SET PERFORM-CHARACTER	MMR-FW30	CMR-CM01
SET PF _n	MMR-FW30	CMR-CM01
SET PREFIX-AREA	MMR-FW30	CMR-CM01
SET PRINT-DELIMITERS	MMR-FW30	CMR-CM01
SET PRINT-SHIFT	MMR-FW30	CMR-CM01
SET REVISION-BARS	MMR-FW30	CMR-CM01
SET SCALE	MMR-FW30	CMR-CM01
SET SCREEN-REFRESH	MMR-FW20	CMR-FE01
SET SCROLL-LIMIT	MMR-FW20	CMR-FE01
SET SOS-DUMP	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
SET SPACE-LINE-DIPLAY	MMR-FW30	CMR-CM01
SET SUBSTITUTION- CHARACTER	MMR-FW30	CMR-CM01
SET TAB-CHARACTER	MMR-FW20	CMR-FE01
SET TABS	MMR-FW20	CMR-FE01
SET TRANSLATION	No equivalent	DYR-UD15
SET UPDATE-OUTPUT	MMR-FW20	CMR-FE01
SET UPPER-CASE	MMR-FW30	CMR-CM01
SET VERIFY	MMR-FW30	CMR-CM01
SFILE	MMR-FW20	CMR-FE01
SHOW ALIAS-TYPES	MMR-FW30	CMR-CM01
SHOW MEMBER-TYPE	MMR-FW30	CMR-CM01
SHOW PRODUCE-OPTIONS	MMR-FW30	CMR-CM01
SHOW UDR	MMR-FW30	CMR-CM01
SHOW UDS	MMR-FW30	CMR-CM01
SIGNAL	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
SKIP	MMR-FW30	CMR-CM01
SMOVE	MMR-FW20	CMR-FE01
SNAPSHOT	MMR-FN24	DSR-DS01
SORT	MMR-FW20	CMR-FE01
SOURCE-FILE	MMR-FW20	CMR-FE01
SPACE	MMR-FW30	CMR-CM01
SQL ACQUIRE	MMR-FN27	DYR-TE32
SQL ALTER	MMR-FN27	DYR-TE32
SQL COMMENT	MMR-FN27	DYR-TE32
SQL CREATE	MMR-FN27	DYR-TE32
SQL DROP	MMR-FN27	DYR-TE32
SQL GRANT	MMR-FN27	DYR-TE32

Name	FU Code(s)	SU Code(s)
SQL LABEL	MMR-FN27	DYR-TE32
SQL LIST CYCLES	MMR-FN24	DSR-PH01
SQL LIST TABLES	MMR-FN24	DSR-PH01
SQL PLOT CLUSTER	MMR-FN24	DSR-PH01
SQL PLOT REFERENTIAL STRUCTURES	MMR-FN24	DSR-PH01
SQL POPULATE	MMR-FN24	DSR-PH01
USING FORMAT	MMR-FN24	DSR-UD30
SQL PREVIEW	MMR-FN24	DSR-PH01
USING FORMAT	MMR-FN24	DSR-UD30
SQL PRODUCE	MMR-FN27	DYR-TE32
SQL REPORT	MMR-FN24	DSR-PH01
USING FORMAT	MMR-FN24	DSR-UD30
SQL REVOKE	MMR-FN27	DYR-TE32
SQL SIZE	MMR-FN27	DYR-TE32
SQL SYNONYM	MMR-FN27	DYR-TE32
SS	MMR-FW20	CMR-FE01
STACK	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
STATUS	MMR-FN30	CMR-AD21 or CMR-DD2
STATUS LIST	MMR-FN30	CMR-AD21 or CMR-DD2
STATUS WINDOW	MMR-FN30	CMR-AD21
STORE KEPT-DATA	MMR-FW30	CMR-CM01
STORE WORKBENCH- DESIGN-AREA	MMR-FN24	DSR-DS01
STRIP	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
SUBSTR	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
SWITCH MESSAGES	MMR-FW30	CMR-CM01
SWITCH NAME- CONCATENATION	MMR-FW30	CMR-CM01

Name	FU Code(s)	SU Code(s)
SWITCH OUTPUT	MMR-FW30	CMR-CM01
TEXEC	MMR-FW30	DYR-TE00
TEXT	MMR-FW30	CMR-CM01
TOP	MMR-FW30	CMR-CM01
TRACE	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
TRANSFER command	MMR-FW30	DYR-DY01
TRANSFER directive	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
TRANSIENT- EXECUTIVE	MMR-FW30	DYR-TE00
TRANSLAT function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
TRANSLATE	MMR-FW30	DYR-DY01
IDD	No equivalent	DYR-TE08
IDMS	No equivalent	DYR-TE08
OTHER-DICTIONARY	MMR-FW30	DYR-TE00
TSS	MMR-FW20	No equivalent
TYPE function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
UP	MMR-FW20	CMR-FE01
UPDATE	MMR-FW20	CMR-FE01
UPPER function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
US	MMR-PF61	No equivalent
USA	MMR-PF61	No equivalent
USAGE	MMR-PF61	one
USR	MMR-PF61	No equivalent
VALIDATE	MMR-PF61	No equivalent
VALUE function	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
VLIST directive	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05

Name	FU Code(s)	SU Code(s)
WAL	MMR-PF61	No equivalent
WBDA	MMR-FN24	DSR-DS01
WF	MMR-PF61	No equivalent
WHAT	MMR-PF61	DMR-DD1
BELONGS-TO	MMR-FW30	DMR-DD3
FATHERS	MMR-FN24	DMR-DD5
FATHERED-BY	MMR-FN24	DMR-DD5
GENERATES	MMR-FN24	DMR-DD5
GENERATED-BY	MMR-FN24	DMR-DD5
RELATED	MMR-PF61	No equivalent
WHICH	MMR-PF61	DMR-DD1
AMENDED	MMR-FN30	CMR-DD2 or CMR-AD21
BELONGS-TO	MMR-FW30	CMR-DD3
CHANGED	MMR-FN30	CMR-DD2 or CMR-AD21
CHECK-NEEDED	MMR-PF61	No equivalent
CHECK-OK	MMR-PF61	No equivalent
CURRENT	MMR-FN30	CMR-DD2 or CMR-AD21
DIVERGING	MMR-FN30	CMR-DD2 or CMR-AD21
FATHERS	MMR-FN24 or MMR-FW25	DMR-DD5
FATHERED-BY	MMR-FN24 or MMR-FW25	DMR-DD5
GENERATES	MMR-FN24 or MMR-FW25	DMR-DD5
GENERATED-BY	MMR-FN24 or MMR-FW25	DMR-DD5
GENERIC	MMR-FW20	CMR-UD1
LOCKED	MMR-FW30	CMR-WS01
NEW	MMR-FN30	CMR-DD2 or CMR-AD21
UNVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
RELATED	MMR-PF61	No equivalent

Name	FU Code(s)	SU Code(s)
REVERIFIED	MMR-FN30	CMR-DD2 or CMR-AD21
WHO	MMR-FW30	CMR-DD3
WHOSE	MMR-PF61	DMR-DD1
WIPE	MMR-FW20	CMR-FE01
WORD	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
WORDINDX	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
WORDLEN	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
WORDS	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
WRITE	MMR-FW20	CMR-FE01
WRITEF	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
WRITEL	MMR-FW30 or MMR-FW20	DYR-TE00 or CMR-UD05
XQUIT	MMR-FW20	CMR-FE01
ZONE	MMR-FW20	CMR-FE01

